공학박사 학위논문

Text-line Detection and Word Segmentation in Document Images based on an Optimization Framework

최적화 방법을 이용한 문서영상의 텍스트 라인 및 단어 검출법

2015 년 8 월

서울대학교 대학원

전기·컴퓨터공학부

유 제 웅

ABSTRACT

Locating text-lines and segmenting words in a document image are important processes for various document image processing applications such as optical character recognition, document rectification, layout analysis and document image compression. Thus, there have been a lot of researches in this area, and the segmentation of machine-printed documents scanned by flatbed scanners have been matured to some extent. However, in the case of handwritten documents, it is considered a challenging problem since the features of handwritten document are irregular and diverse depending on a person and his/her language. To address this problem, this dissertation presents new segmentation algorithms which extract text-lines and words from a document image based on a new super-pixel representation method and a new energy minimization framework from its characteristics.

The overview of the proposed algorithms is as follows. First, this dissertation presents a text-line extraction algorithm for handwritten documents based on an energy minimization framework with a new super-pixel representation scheme. In order to deal with the documents in various languages, a language-independent text-line extraction algorithm is developed based on the super-pixel representation with normalized connected components(CCs). Due to this normalization, the proposed method is able to estimate the states of super-pixels for a range of different languages

and writing styles. From the estimated states, an energy function is formulated whose minimization yields text-lines. Experimental results show that the proposed method yields the state-of-the-art performance on various handwritten databases.

Second, a preprocessing method of historical documents for text-line detection is presented. Unlike modern handwritten documents, historical documents suffer from various types of degradations. To alleviate these problems, the preprocessing algorithm including robust binarization and noise removal is introduced in this dissertation. For the robust binarization of historical documents, global and local thresholding binarization methods are combined to deal with various degradations such as stains and fainted characters. Also, the energy minimization framework is modified to fit the characteristics of historical documents. Experimental results on two historical databases show that the proposed preprocessing method with text-line detection algorithm achieves the best detection performance on severely degraded historical documents.

Third, this dissertation presents word segmentation algorithm based on structured learning framework. In this dissertation, the word segmentation problem is formulated as a labeling problem that assigns a label (intra-word/inter-word gap) to each gap between the characters in a given text-line. In order to address the feature irregularities especially on handwritten documents, the word segmentation problem is formulated as a binary quadratic assignment problem that considers pairwise correlations between the gaps as well as the likelihoods of individual gaps based on the proposed text-line extraction results. Even though many parameters are involved in the formulation, all parameters are estimated based on the structured SVM framework so that the proposed method works well regardless of writing styles and written languages without user-defined parameters. Experimental results on IC-

DAR 2009/2013 handwriting segmentation databases show that proposed method achieves the state-of-the-art performance on Latin-based and Indian languages.

Key words: document image segmentation, text-line extraction, word segmentation, energy minimization framework, structured learning, super-pixel representation, historical document

Student number: 2009-20798

Contents

List of Figures

List of Tables

Chapter 1

Introduction

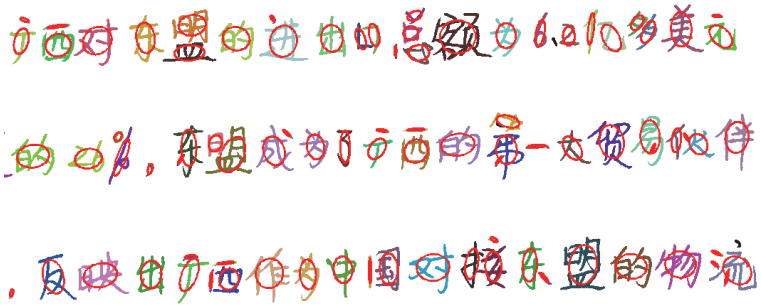
Segmentation of document images into text-lines and words is an essential step for various document image processing tasks such as layout analysis, rectification of documents, optical character recognition (OCR) and document image compression. Hence, there have been a lot of researches in this area, and a number of algorithms have been proposed for the extraction of text-lines [1–3] and words [4, 5] in machine-printed document images. In the case of machine-printed documents, it is considered a solved problem when documents are read by flatbed scanners since the structure of text-lines and words is relatively regular and well structured. However, segmentation of handwritten documents into text-lines and words is still considered a challenging problem because the scale and orientation of characters are spatially varying, spacings between text-lines and words are irregular, characters may touch across words and/or text-lines and there are variations of writing styles depending on the person. This problem is worse in the case of historical documents which suffers from degradations. In order to address these problems, many algorithms have been developed in the past few decades for text-line extraction [6–16] and word

segmentation [14–25] for handwritten documents.

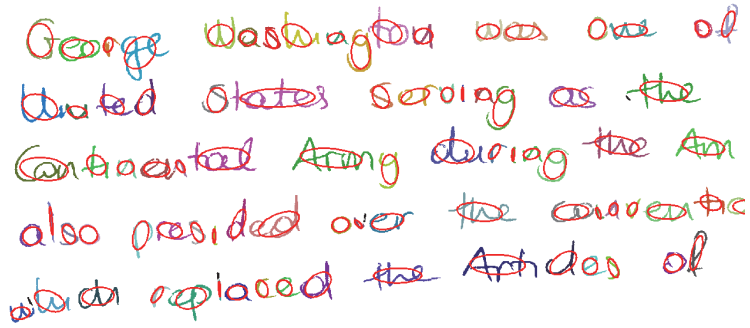
In this dissertation, text-line extraction algorithm and word segmentation algorithm via energy minimization framework are proposed. Unlike previous algorithms for machine-printed and handwritten document images, the proposed algorithms are able to deal with handwritten documents regardless of their languages as well as machine-printed documents. Also, it can be applied to the text-line extraction of historical documents with appropriate preprocessing methods. To achieve these objectives, a new super-pixel representation technique to describe characters in document images based on their stroke width is proposed. Using proposed super-pixel representation method, each component of given document images is well-represented regardless of its contents. Then, energy functions for solving text-line extraction and word segmentation are designed and optimized, which yields the segmentation results.

1.1 Text-line Detection of Document Images

There have been many approaches to text-line extraction of document images. For example, there are many techniques such as Hough transform [7], active contours [8], Hidden Markov Model [9], and energy minimization formulation [11]. However, most conventional work focused on specific character sets. That is, conventional algorithms address the variations caused by individual writers by exploiting language-specific features. For example, in Chinese documents, two characters are usually separated as shown in Fig. 1.1-(a). Thus, conventional methods are based on connected component (CC) analysis [6, 11]: they extract CCs and partition them into text-lines. On the other hand, Latin-based scripts (e.g., English, German, and Greek



(a)



(b)

(c)

Figure 1.1: Connected component representation of various languages. (a) Chinese script. (b) English script. (c) Bangla (traditional Indian) script.

documents) and Bangla (the second language of India) have different characteristics. Unlike Chinese scripts, many graphemes in these scripts are connected due to cursive writing as shown in Fig. 1.1-(b) and (c), making these CC-based approaches fail. To be specific, the approach in [11] estimated spatially varying states (line spacing and orientation) from the distribution of CCs, and it has problems in dealing with cursive Latin-based scripts. The situation is worse for Indian scripts where most characters are connected (Fig. 1.1-(c)). On the other hand, character components are placed in a one-dimensional way in Latin-based and Indian scripts as illustrated in Fig. 1.1-(b) and (c), allowing us to develop horizontal bottom-up clustering rules [13, 26]. However, this bottom-up approach may not work for Chinese scripts, where character components are placed in a two-dimensional way as in Fig. 1.1-(a).

To develop a language independent algorithm, an extended approach from [11] is proposed in this dissertation. In [11], a cost function whose minimization yields text-lines has been developed. However, it fails when only a small number of connected components are available, which is a common situation for cursive writing of Latin-based languages and Indian handwritten scripts as shown in 1.1-(b) and (c). In order to overcome this limitation, a new super-pixel representation method is developed by partitioning under-segmented CCs into normalized ones. By the proposed normalized super-pixel method, the scales and orientations of each character component can be estimated reliably in a variety of documents. However, this idea could introduce problems in the energy minimization, because the connectivity information is sometimes lost (due to the partitioning). Hence, the energy minimization method is also improved to address this connectivity loss problem. Experimental results show that the proposed method yields the best performance on various handwritten

databases: HIT-MW database in Chinese [27], ICDAR 2009/2013 handwriting segmentation contest database in Latin-based/Indian languages [28, 29], IAM database in English [30] and UMD database in Arabic [31].

In this dissertation, an extended approaches of text-line extraction algorithm to degraded historical documents are also proposed. For historical documents, this dissertation presents preprocessing algorithms that make the historical documents be applicable to text-line segmentation algorithm. Text-line detection performance on historical documents is evaluated on 2 databases: George Washington database [32] and training set of competition on ICDAR 2015 text-line detection of historical documents [33].

1.2 Word Segmentation of Document Images

The word segmentation problem has been considered a subproblem of text-line segmentation since the structure between text-line and word is hierarchical. Therefore in many researches [14–25], document images are first segmented into text-lines by text-line segmentation methods [6, 11, 34]. Then, the word segmentation algorithm for a single text-line is applied to individual text-lines to label the words in a text-line. Given a single text-line results by text-line segmentation algorithm, the conventional word segmentation algorithms usually consist of two steps: the first step is to extract candidates for inter-word gaps (word-separator) and the next step is to classify the candidates into intra-word gaps and inter-word gaps as shown in Fig. 1.2. For the generation of candidate gaps as in Fig. 1.2, a given text-line is represented with a set of super-pixels (where a super-pixel usually corresponds to a letter or a group of letters). Then, the gaps between these super-pixels are

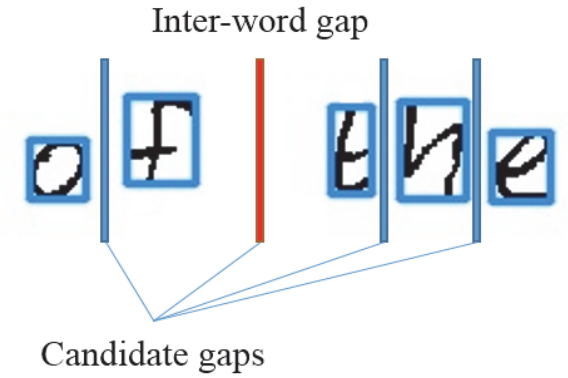


Figure 1.2: An illustration of candidate gaps and inter-word gaps in a single text-line. The boxes represent super-pixels and the bars between the boxes are the candidates gaps. Blue/red bars mean intra-word/inter-word gap by classification algorithm, respectively.

considered candidates to be classified. This can be seen as a binary classification problem that assigns a label from the set of $\{0, 1\}$, where 0 means that the gap is an intra-word gap and 1 indicates it is an inter-word gap.

For the machine-printed documents, this classification is considered a solved problem since gap widths of intra-word and inter-word are significantly different [25, 35]. Therefore, it can be solved easily with adaptive thresholding and clustering. However, unlike machine-printed documents, the segmentation of handwritten documents is still considered a challenging problem due to (i) the irregular spacings between words and (ii) the variations of writing styles depending on the person. Thus, for handwritten documents, many algorithms have been developed. The global/adaptive thresholding by the characteristic of candidate gaps was popularly

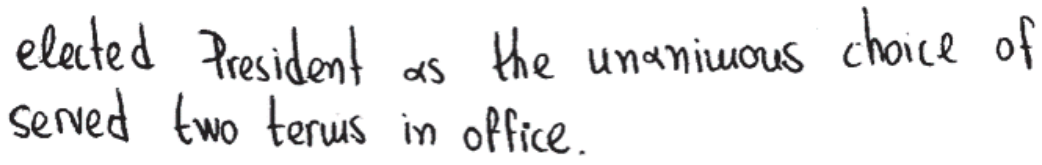


Figure 1.3: An example shows the correlation of inter-word gaps within a text-line. The widths of inter-word gaps in the first text-line are larger than those of the second text-line and their widths are similar within each text-line.

used to classify them as in [15–18] for a few decades. Also, the unsupervised learning techniques such as clustering and Gaussian Mixture Model (GMM) were adopted in [14, 19]. The scale space selection approach was employed in [22] and there have been researches using supervised-learning techniques such as neural networks [23,24]. However, they considered only the local properties of individual gaps (without the considerations on correlations between the gaps).

Although the characteristics of inter-word gaps are changing across (and even in) documents, there are strong correlations (e.g., scale, width) between them within a text-line as shown in Fig. 1.3. However, it has been difficult to exploit these correlations in the conventional approaches, where the classification is made independently based on the properties of each gap as in previous researches [14–24]. In order to alleviate these problems, a new framework that considers these correlations between the candidate gaps as well as local observations (i.e., the properties of each gap) is proposed in this dissertation. To be precise, the word segmentation problem is formulated as an optimization problem that maximizes the similarity between inter-word gaps and the dissimilarity between inter-word and intra-word gaps, in ad-

dition to the local likelihoods. Since this problem is a binary classification problem and the singleton and pairwise terms are only considered, it can be formulated as a binary quadratic problem, which can be efficiently solved with the Mixed-Integer Quadratic Programming (MIQP) solvers. Also, all parameters are estimated by adopting the structured learning framework [36], so that the proposed method can deal with a variety of inputs without user-defined parameters. Experimental results on ICDAR 2009 and 2013 handwriting segmentation contest database [28, 29] show that proposed algorithm yields the best performance on both databases. In addition to improved performances, the main contributions of this dissertation on word segmentation can be summarized as follows: (i) a new formulation of the word segmentation problem into a binary quadratic problem and (ii) a principled approach to the parameter learning based on Structured SVM.

1.3 Summary of Contribution

Locating text-lines in machine-printed document images and segmentation of words from them are now considered solved problems when the image is read by flatbed scanners. However, these problems for handwritten and historical documents are still considered challenging problems. This dissertation presents a new framework focused on the text-line and word segmentation of handwritten documents, which can also work for the machine printed and historical document images with proper preprocessing method. For this purpose, this dissertation makes some contributions which can be summarized as

- Development of new super-pixel representation method by normalizing the size of connected component(CC) with respect to stroke width information

- Formulation of energy equations whose optimization yields the text-line and word segmentation results for various kinds of documents
- A binary quadratic problem formulation for word segmentation and parameter learning with Structured SVM
- The state-of-the-art performance on various databases.

Chapter 2

Related Work

The segmentation of document images has been researched over several decades [1,4]. Especially on handwritten documents, the segmentation of them have received high attention for text-lines [37] and words [17] due to its many difficulties. Also, several competitions have been organized by ICDAR [28, 29, 33, 38] for improving the performance of segmentation on handwritten document images. In this chapter, previous methods for the segmentation of text-lines and words are reviewed.

2.1 Text-line Detection

Previous researches for the text-line detection can be classified into 4 categories [37, 39]: projection-based, Hough transform-based, bottom-up grouping-based and graph-based methods.

Projection based methods are based on the vertical projection profiles and popularly exploited for the text-line detection of machine-printed documents [40]. In this approach, text-line separator is determined by analyzing valleys and peaks of

the projection profiles. Due to its efficiency and simplicity, many text-line detection methods based on projection profile [15, 16, 22, 41]. However, the projection-based methods do not work well for handwritten documents since the text-lines are curved and many characters are touching across the text-lines.

Hough-based methods [7, 42, 43] exploited Hough transform to detect the text-line of handwritten documents. In [42], they first represent characters by connected components (CC) and apply some preprocessing algorithm to separate the CCs. Then, Hough transform is used to extract each text-line. Although this approach is able to handle the variable skew angle of each text-line, it cannot deal with the curved one whose skew angle is varying within the text-line.

Bottom-up grouping methods [1,6,11,44] merge the adjacent CCs into text-lines so that they can handle the curved text-lines. In this approach, merging rules are important for the performance of text-line segmentation. For example, in [6], they constructed tree structure of CCs for document segmentation and merged the CCs with minimal spanning tree algorithm. However, it has some problems with heuristic merging rules and suffers from the CCs touching across text-lines. To address these problems, energy minimization framework is adopted for the grouping of CCs into text-lines in [11]. Also, they solve the problem of touching across characters by exploiting fitting functions of clusters. However, they have some problems when the most of characters in the scripts are connected.

In graph-based approaches [39,45], the text structure is represented by the graphs and energy functions are designed to find the optimal path between the text-lines. In [45], projection profiles are combined with graph-based method for the robust text-line segmentation. In [39], they formulate the skeleton of the background image as a graph. Then, an optimum path between the text-lines is found by path-finding

algorithm which give the path that has the lowest cost.

2.2 Word Segmentation

Word segmentation is often formulated as a gap classification problem which labels each gap into intra-word and inter-word gap in a given text-line [17]. Therefore, it is categorized into classification method of the gaps between the characters.

The most popular approach of word segmentation is thresholding [15–18]. This approach is based on the observation: gap width of the inter-word is significantly larger than that of intra-word. Thus, threshold of determining inter-word and intra-word gap is estimated by histogram of SVM-based gap metrics in a given line [16] and by heuristic rules [17, 18]. The threshold-based classification is very efficient and easy to implement. Also, it has good classification performance on clean handwritten scripts as in machine-printed document. However, this method cannot deal with when the irregular spacings exist between characters which are common in cursive handwritten scripts. Further, its performance is strongly depends on the gap distance metrics.

Another approach is based on the unsupervised clustering algorithms [19, 42]. Gaussian Mixture Model by EM algorithm of gap metrics is adopted in [42]. They exploited overlapped connected components(OCs) to represent character and convex hull-based metric as a distance measure. In [19], sequential clustering of three different gap metrics is used to classify gaps.

On the other hand, there are some researches which are not included by gap classification-based problem. In [22], scale space is adopted for the word segmentation problem. They apply 2D Gaussian filters with different σ_x, σ_y to a given

text-line image and find the optimal scales which representing the word entities well. Also, there is a research which adopts the simple neural network for the word segmentation [23]. In the work, they first get the separate characters by character segmentation algorithm. Then, simple neural network is adopted with 8 properties such as intervals and heights of characters.

Chapter 3

Text-line Detection of Handwritten Document Images based on Energy Minimization

3.1 Proposed Approach for Text-line Detection

Text-line extraction can be considered a super-pixel segmentation (grouping) problem and the proposed method is based on this observation: extract super-pixels which represent character components and estimate their spatially-varying states(scales and orientation) then adopt bottom-up grouping into text-line with energy minimization framework. In this chapter, a new super-pixel representation method based on normalizing connected components(CCs)is presented. Also, text-line segmentation framework based on an energy minimization is explained.

3.1.1 State Estimation of a Document Image

Since the scales and orientations of characters (text components) differ between the documents, a state (scale and orientation) of each super-pixel is needed to be estimated for text-line segmentation. The idea of estimating the scale and orientation (i.e. state) for a document image processing have been proposed [1, 2]. In docstrum [1], they exploited histograms (angle and nearest neighbor distances) to estimate the orientation, inter-line spacing and within-line spacing. Using the state information, they adopted bottom-up grouping method to cluster super-pixels into several components such as words, text-lines and paragraphs. However, this approach introduces problems in handwritten documents where these states are not fixed (spatially-varying) even in a single document since it assumes that the states are fixed in a whole document.

To address these problems, the spatially-varying states of super-pixels for a given document are needed to be estimated. This idea was first introduced in [3], where the states were estimated based on the distributions of super-pixels (CCs). A review of state estimation algorithm is followed. In [3], they assume that each super-pixel has its own state. Thus, for every super-pixel at site $p \in \mathcal{P}$, state $g_p = (s_p, \theta_p)$ is assigned, where s_p is the interline spacing between text-lines and θ_p is the orientation of text-line in super-pixel at site p. The state estimation problem is formulated as an energy minimization problem given by

$$E(g_p) = \sum_{p \in \mathcal{P}} V_D(g_p) + \sum_{(p,q) \in \mathcal{E}} V_P(g_p, g_q), \tag{3.1}$$

where $\mathcal{E}$ is a set of edges, $V_D(g_p)$ is a data term reflecting local properties and $V_P(g_p, g_q)$ is a pairwise term for smoothness.

For the design of $V_D(g_p)$, super-pixels are represented as ellipses by computing

the mean vector and the covariance matrix. Then, a projected signal is defined as the number of ellipses in the direction of the projection. If super-pixels located in the neighborhood of a site p are projected to the normal direction to a text-line, a periodic pattern whose period is interline spacing (s_p) appears in the projected signal. By exploiting these properties, $V_D(g_p)$ is designed to be decreased when its periodicity of projected signal is increasing.

From these observations, a projected signal $x(n)$ is obtained by projecting super-pixels around a site p to the orientation θ_p. To measure the periodicity, its DFT (Discrete Fourier Transform) $X_N(k)$ is computed as

$$X_N(k) = \sum_{n=0}^{N-1} x(n)\exp^{-j\frac{2\pi kn}{N}}. \tag{3.2}$$

Then, the normalized energy of periodicity with period $\frac{N}{k}$ can be approximated by

$$\frac{|X_N(k)|^2 + |X_N(2k)|^2 + \cdots}{|X_N(0)|^2 + |X_N(1)|^2 + |X_N(2)|^2 + \cdots} \simeq \frac{|X_N(k)|^2}{|X_N(0)|^2}. \tag{3.3}$$

The numerator means the energy of repeating component with $T = \frac{N}{k}$ and the denominator is the signal energy of $x(n)$. Based on the normalized energy of periodicity, the data term $V_D(g_p)$ is defined as

$$V_D(g_p) = -\log \frac{|X_{N_p}(k_p)|^2}{|X_{N_p}(0)|^2}. \tag{3.4}$$

For the final step, an appropriate set of (s_p, N_p, k_p) has to be decided which satisfies $\frac{N_p}{k_p} = s_p$. If N is large, the resolution of frequency domain is increased. However, detection of spatially varying state is difficult with large N because it covers wide area around site p to detect periodicity. Considering these factors including computational complexity, 10 scales s_p from 12.8 to 128 are selected. Also, orientation θ is quantized into 32 steps like

$$\theta_p \in \left\{ i \times \frac{\pi}{32} \mid i = 0, \ldots, 31 \right\}. \tag{3.5}$$

For smoothness cost, a neighborhood system is constructed by Delaunay triangulation [46]. Then, a neighborhood term is given by

$$V_P(g_p, g_q) = \rho(g_p, g_q) \times \exp\left(-\frac{k \times d_{pq}^2}{(s_p^2 + s_q^2)}\right), \tag{3.6}$$

where d_{pq} is an Euclidean distance between two site p and q. $\rho(g_p, g_q)$ is the label discontinuity function which allows small amount of label discontinuities defined as

$$\rho(g_p, g_q) = \begin{cases} 0 & g_p = g_q \\ \lambda_1 & |g_p - g_q| \leq 3 \\ \lambda_2 & \text{otherwise} \end{cases} \tag{3.7}$$

where $|g_p - g_q|$ is the label distance. To optimize (3.1), *Expansion Move* algorithm [47] is adopted.

3.1.2 Problems with Under-segmented Super-pixels for Estimating States

As stated above section, an energy minimization problem is formulated to estimate the state(inter-line spacing and orientation) of each super-pixel. This idea was also adopted in the text-line extraction problem of handwritten Chinese documents [11]. In the work, to deal with the bottom-up grouping errors by irregularly distributed super-pixels and the curvilinear text-lines of the handwritten documents, the cost function which considers fitting errors of text-lines as well as interline distances was developed based on the estimated states of super-pixels. Then, the cost function was minimized by applying small variations to its coarse solution, which yields text-line segmentation results. It was successful in extracting text-lines in handwritten Chinese documents.

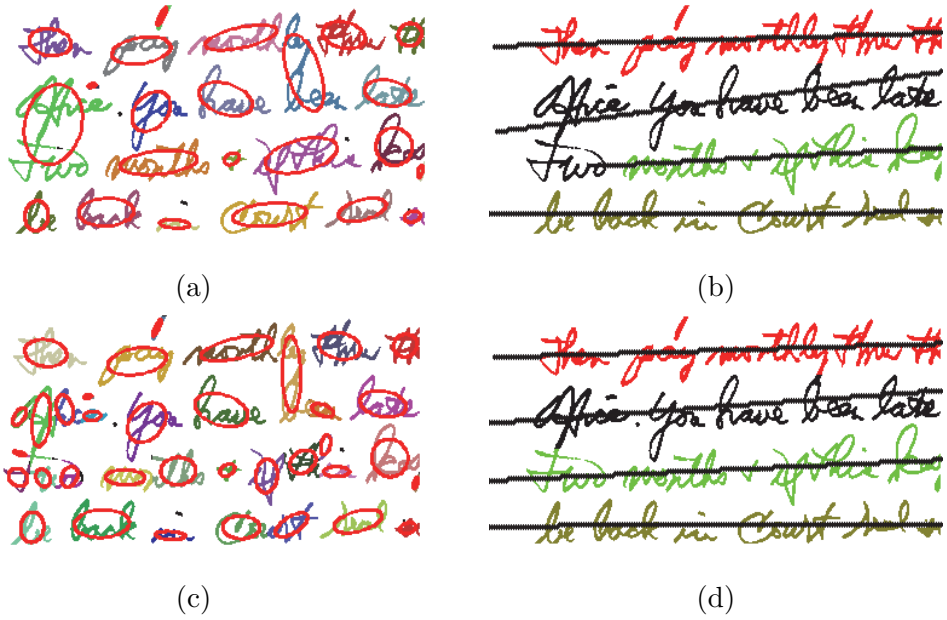


Figure 3.1: (a) Conventional super-pixel representation (connected components) by the method in [11]. (b) Text-line extraction result of [11]. (c) Super-pixel representation of proposed method. (d) Text-line extraction result of proposed method. In the above representation, each color represents an individual super-pixel and red ellipses are the approximations of super-pixels.

However, when the size of a super-pixel is too large and/or there are only a small number of super-pixels, the spatially varying states cannot be correctly estimated since the data term $V_D(g_p)$ of state estimation method is the function of the number of super-pixels around a given location. Moreover, connected characters by touching across the neighboring text-lines introduce problems in CC grouping (i.e., text-line extraction) as shown in Fig. 3.1-(a) and (b). Therefore, the previous method [11] cannot work for Latin-based, Indian and Arabic handwritten documents where above-mentioned cases are common. In order to address these problems, a new super-pixel extraction method that partitions under-segmented CCs into normalized ones as shown in Fig. 3.1-(c) is proposed. This idea makes the algorithm estimate the spatially-varying states even for documents having under-segmented CCs as illustrated in Fig. 3.1-(d). Therefore, it can be applied for a range of languages(Latin-based, Chinese, Bangla and Arabic), document types(handwritten, historical, machine-printed documents) and writing styles. One might think that character segmentation algorithms [48] could be adopted for these documents. However, character segmentation is a challenging problem itself and does not work in a language-independent manner.

3.1.3 A New Super-pixel Representation Method based on CC Partitioning

As mentioned in Section 3.1.2, a new super-pixel representation method is required to estimate the spatially-varying states of super-pixels correctly in any kinds of documents. Thus, this section presents the method to partition CCs into sub-segments so that they have normalized sizes.

In order to make the algorithm work in a scale-invariant manner (i.e., yield the

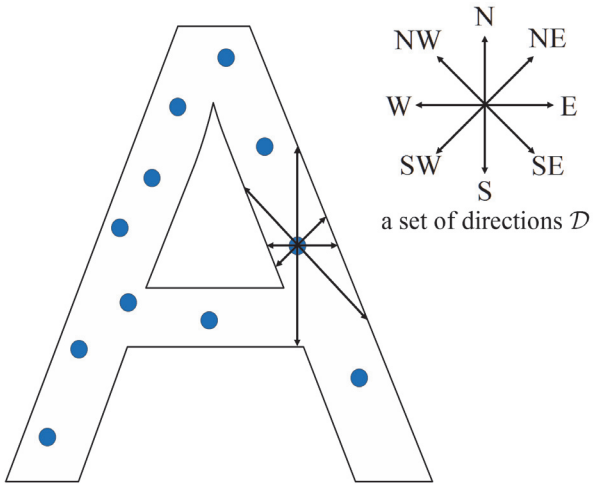


Figure 3.2: Illustration of our stroke-width estimation method. Blue dots represent randomly-selected seed points.

same results regardless of the scanning resolution of input documents), a stroke length is exploited. Intuitively, the stroke length represents how far the pen moves to write a given character represented as a CC. Therefore, it can be exploited as a threshold to decide under-segmented CC. For the presentation of proposed method, assume that a CC denoted as C_j, with its stroke width W_j [49]. Since the accurate estimation of W_j is not essential in this approach, an efficient estimation algorithm is proposed. To estimate stroke width of given CC, n random pixels in C_j are selected to build a set of seed points $\mathcal{S}$

$$\mathcal{S} = \{p_i\}_{i=1}^{n}, \tag{3.8}$$

where $p_i \in \Re^2$. After constructing a set $\mathcal{S}$, four rays travel across C_j starting from $p_i (i = 1, \ldots, n)$ in four directions N-S, W-E, NW-SE, NE-SW as shown in Fig. 3.2.

Then, the estimated stroke width is defined as the mean of the minimum distances:

$$W_j = \frac{1}{n}\sum_{i=1}^{n} \min_{d\in\mathcal{D}} (W_d(p_i)), \tag{3.9}$$

where $\mathcal{D} = \{\text{N-E, W-E, NW-SE, NE-SW}\}$ is a set of directions and $W_d(p_i)$ is a width along the direction $d \in \mathcal{D}$. Using W_j, the stroke length of C_j is defined as

$$L_j = \frac{|C_j|}{W_j}, \tag{3.10}$$

where $|C_j|$ is the number of pixels in C_j.

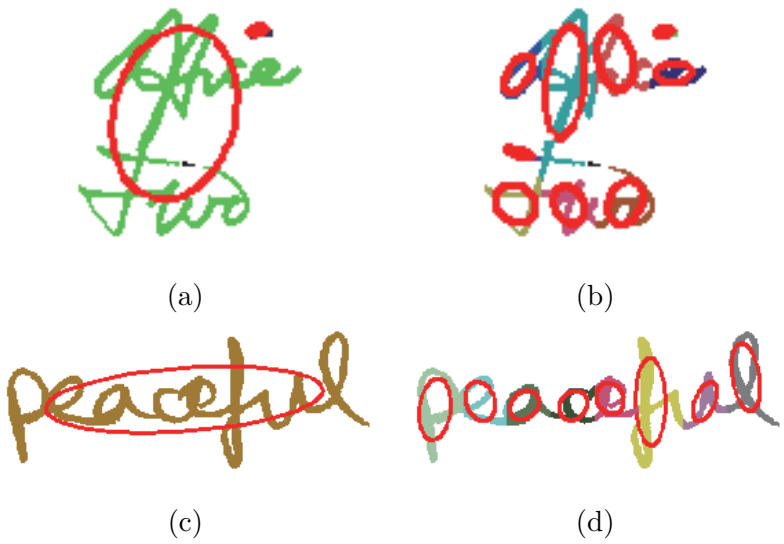
To normalize the size of CCs to get the proposed super-pixel representation, the under-segmented CCs have to be decided and partitioned into several components to have appropriate sizes. For this, C_j is considered to be under-segmented when its (normalized) stroke length is higher than a threshold:

$$L_j > \alpha \times \overline{W}. \tag{3.11}$$

where $\overline{W}$ is the mean of stroke lengths in a given image. Note that $\frac{L_j}{\overline{W}}$ is a scale-invariant measure for stroke length. For CCs satisfying (3.11), they are partitioned so that the width of each segment becomes

$$\beta \times \overline{W}. \tag{3.12}$$

In other words, proposed super-pixel extraction algorithm to partition CCs into normalized ones consists of two steps: (i) to select CCs that should be segmented (i.e. under-segmented CC) and (ii) to partition selected CCs into smaller ones. The parameter α in (3.11) controls the minimum size of CCs to be segmented (larger α makes less CCs to be partitioned), and the parameter β in (3.12) determines the size of partitioned CCs. Since these parameters work in a scale-invariant manner (due to $\overline{W}$), one set of fixed parameters can be used on various databases.



(a) (b)

(c) (d)

Figure 3.3: Elliptical approximation of super-pixels. (a),(c) Conventional CC-based Super-pixel representation method by [3]. (b),(d) Super-pixel representation after applying the proposed CC partitioning method. Different colors are used for individual super-pixels.

As shown in Fig. 3.3, the proposed method yields regularized super-pixels and this enables the correct extraction of text-lines as presented in Fig. 3.1-(b) and (d). This CC partitioning is an important step in the proposed algorithm because the spatially-varying line spacings and orientations of super-pixels are estimated based on [3]. To be precise, this method [3] estimated the states of super-pixels from their distribution by assigning an equal weight to every super-pixel. That is, the projection profile is not proportional to the sizes of super-pixels but to the number of super-pixels. Therefore, this approximation works only when (i) there are a sufficient number of super-pixels in a neighborhood and (ii) super-pixels have

similar sizes (i.e. similar width/height). However, neither of them is satisfied by the super-pixels in cursive Latin-based or Indian scripts. In order to handle such scripts as well as machine-printed and Chinese scripts, the proposed super-pixel representation method is exploited to satisfy both conditions as illustrated in Fig. 3.1 - (c).

3.1.4 Cost Function for Text-line Segmentation

Based on the normalized CCs in the previous section, the line spacing and orientation of every super-pixel are estimated with the method in [3]. From the estimated states, the cost function for text-line extraction is the same as that of [11], given by

$$(\hat{K}, \hat{\Lambda}) = \arg\min_{K,\Lambda} \left(E_L(\Lambda) + E_G(\Lambda) \right), \tag{3.13}$$

where $\hat{K}$ denotes the number of detected text-lines and $\hat{\Lambda}$ represents detected $\hat{K}$ text-lines. Precisely,

$$\hat{\Lambda} = \{\Lambda_1, \Lambda_2, \ldots, \Lambda_{\hat{K}}\}, \tag{3.14}$$

where Λ_i is a set of super-pixels labeled as the k-th text-line and $\Lambda_i \cap \Lambda_j = \emptyset (0 \leq i \leq j \leq \hat{K})$. A review of the designing the cost function (3.13) is followed.

In (3.13), the cost function is consisted of two terms, $E_L(\Lambda)$ and $E_G(\Lambda)$. $E_L(\Lambda)$ is the local cost that derived as a function of normalized fitting errors of text-lines. The fitting function of a labeled k-th text-line (cluster) Λ_k is given by

$$f_{\Lambda_k} = \arg\min_{f \in P[x]} \sum_{l \in \Lambda_k} (y_l - f(x_l))^2, \tag{3.15}$$

where $P[x]$ is a set of the nth-order polynomials and (x_l, y_l) is the center coordinate of l-th super-pixel in Λ_k. Then, the fitting error of a Λ_k can be calculated by a

root-mean-square error

$$\epsilon(\Lambda_k) = \sqrt{\frac{1}{|\Lambda_k|} \times \sum_{l \in \Lambda_k} (y_l - f_{\Lambda_k}(x_l))^2} \tag{3.16}$$

where $|\cdot|$ is the operator returns the number of elements(pixels). The order of fitting function f_{Λ_k} in (3.15) is determined by the type of documents and the size of cluster $|\Lambda_k|$. For the handwritten and historical documents obtained by flatbed scanner, there is no perspective distortion thus almost all text-lines are straight except some curved text-lines by writing style. Therefore, the order of fitting functions in those documents is set from 0 to 2 with respect to the size of cluster(the number of super-pixels in a cluster Λ_k), i.e. assign higher order when the size of cluster is large. In order to make $\epsilon(\Lambda_k)$ be scale-invariant, it is normalized as

$$\epsilon_n(\Lambda_k) = \frac{\epsilon(\Lambda_k)}{s(\Lambda_k)} \tag{3.17}$$

by the estimated line spacing of Λ_k

$$s(\Lambda_k) = \frac{1}{|\Lambda_k|} \sum_{l \in \Lambda_k} s_l. \tag{3.18}$$

Since s_l is the estimated line spacing of l-th super-pixel, the estimated line spacing of around Λ_k can be calculated by taking the average of the line spacings of super-pixels in Λ_k. Based on the normalized fitting error (3.17) and the small fitting error is desirable, $E_L(\Lambda_k)$ is given by

$$E_L(\Lambda) = \sum_{\Lambda_k \in \Lambda} \phi(\epsilon_n(\Lambda_k)) \tag{3.19}$$

with non-decreasing function $\phi(x)$. To design ϕ, they found the most text-lines satisfies $\epsilon_n(\Lambda_k) < 0.2$ (curvilinear text-lines) and the cost difference of two curvilinear text-lines should be small. Thus, they set

$$\phi(x) \propto \exp^{-\frac{1}{x}} \tag{3.20}$$

and the scale is selected to satisfy

$$\phi(0.25) = 1. \tag{3.21}$$

Therefore, the cost function $E_L(\Lambda)$ stays flat when $\epsilon_n(\Lambda_i) < 0.2$.

The second term of $E_G(\Lambda)$ (3.13) represent the inter-line distance between text-lines, i.e. the minimum distance between two fitting functions of corresponding text-lines. However, in some cases, the domain of fitting functions of text-lines (range of x) is not the same. To address this problem, they define the distance between two text-lines (i) when two text-lines have common interval and (ii) when two text-lines have no common interval. First, they define the interval of k-th text-line Λ_k as

$$I(\Lambda_k) = \left[\min_{l \in \Lambda_k} x_l - \delta, \min_{l \in \Lambda_k} x_l + \delta\right] \tag{3.22}$$

where δ is mean width of the super-pixels in Λ_k. The distance is defined as

$$\text{dist}(\Lambda_i, \Lambda_j) = \min_{x \in I(\Lambda_i) \cap I(\Lambda_j)} |f_{\Lambda_i}(x) - f_{\Lambda_j}(x)| \tag{3.23}$$

when Λ_i and Λ_j have the same interval. Otherwise, the distance between two text-lines is given by

$$\text{dist}(\Lambda_i, \Lambda_j) = |f_{\Lambda_i}(x'_1) - f_{\Lambda_j}(x'_2)|, \tag{3.24}$$

where (x'_1, x'_2) is the closest pair on x-axis. They normalize the distance as (3.17) by exploiting the estimated line spacings as

$$\text{dist}_\text{n}(\Lambda_i, \Lambda_j) = \frac{dist(\Lambda_i, \Lambda_j)}{\min(s(\Lambda_i), s(\Lambda_j))}. \tag{3.25}$$

In (3.13), $E_G(\Lambda)$ reflect that two text-lines should not be too close. Thus they apply a soft-threshold function $\gamma(\cdot)$ like

$$E_G(\Lambda) = \sum_{\Lambda_i, \Lambda_j \in \Lambda} \gamma(\text{dist}_\text{n}(\Lambda_i, \Lambda_j)) \tag{3.26}$$

where

$$\gamma(x) = 1 - \tanh(c \times (x - x_0)). \tag{3.27}$$

The values of parameters of soft thresholding are set to $x_0 = 0.5$ and $c = 10$.

As stated before, $E_L(\Lambda)$ is a function of normalized fitting errors of text-lines and $E_G(\Lambda)$ reflects the inter-line distance. Intuitively, $E_L(\Lambda)$ becomes small when the centers of CCs in $\Lambda_i(1 \leq i \leq \hat{K})$ yield a small error to a lower-order polynomial fitting function. On the other hand, $E_G(\Lambda)$ becomes small when the distance between two text-lines (fitting functions) is large. Since these two terms are complementary, accurate text-lines can be extracted by minimizing above energy equation (3.13).

3.1.5 Minimization of Cost Function

The cost function in (3.13) is non-convex thus its minimization consists of two steps [11]. First, a coarse solution is obtained by using a bottom-up grouping method in [3]. In this step, a rectangle for each super-pixel is drawn with its states whose size is $ws_p + hs_p$, center coordinate is (x_p, y_p) and θ_p rotation angle. Then, a new cluster is developed by merging the different rectangles which overlaps. Due to the variant scales of spacing of documents, a single set of parameters (w, h) is not appropriate for this step. Thus, parameter sequence is built with following incrementally increasing sequence: $w_1 = 0.4, w_2 = 0.6, w_3 = 0.8, w_4 = 1.0, w_5 = 1.2$, with $h_1 = 0.22, h_2 = 0.22, h_3 = 0.2, h_4 = 0.2, h_5 = 0.2$. The merging rule of two super-pixels into a text-line follows:

1. Two super-pixels(clusters) are connected with a rectangle constructed by new w_i.

2. The overlapping horizontal length(x domain) is shorter than 10% of their total length.

3. The merged cluster($\Lambda_p \cup \Lambda_q$) still satisfies curvilinearity, i.e. the normalized fitting error should be smaller than 0.2 ($\epsilon_n(\Lambda_k) < 0.2$).

With this method, the coarse solution is obtained effectively by bottom-up grouping technique which exploits the estimated states. For the machined-printed documents by flatbed scanner or camera, this bottom-up grouping results with some refinement steps can be a final solution since the structure of the super-pixels is relatively regular.

However, a lot of small clusters are generated especially in handwritten documents. Thus, it is refined iteratively by applying small variations (proposals) to the current solution. That is, starting from the coarse solution, the minimization is performed in a greedy manner. In [11], four proposals were used:

1. *merge* move

2. *split* move

3. *merge-split* move

4. *merge-merge-split* move

For example, *merge* proposal merges two neighboring text-lines into one(and compute a new fitting function). They implemented *split* proposal in two ways: (i) split under-segmented text-line into two lines by fitting function that runs through the spaces between them. (ii) split text-line by exploiting the valley of projection profile which representing the gap between two text-lines(for relatively straight lines). The

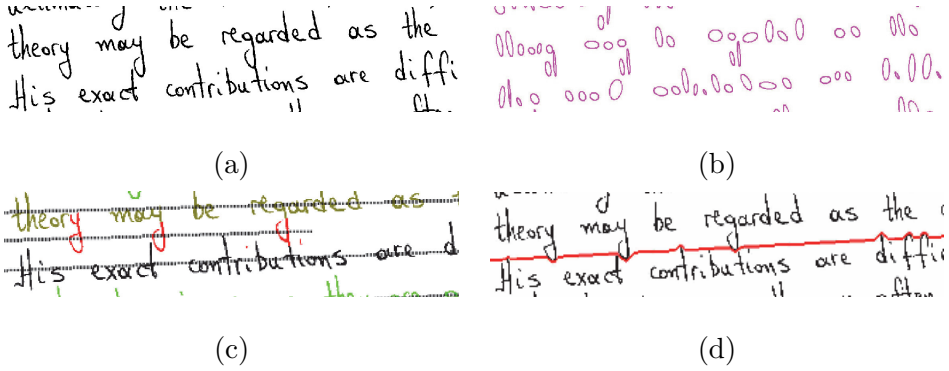


Figure 3.4: (a) Input document. (b) Our super-pixel representation. (c) Floating super-pixels may introduce problems in text-line extraction. (d) New text-line split proposals based on dynamic programming.

third and fourth moves are combination of *merge* and *split* proposals. For example, *merge-merge-split* move first merges three neighboring clusters and split the merged cluster into two clusters.

However, unlike Chinese document cases [11], the split proposals do not work well in documents written in Latin-based, Indian and Arabic languages. This is because (i) the space between two adjacent lines is less clear in Latin-based documents and (ii) the proposed CC partitioning method may result in floating super-pixels between text-lines as shown in Fig. 3.4-(a) and (b). In other words, when under-segmented CCs are partitioned into normalized super-pixels as shown in Fig. 3.4-(b), previous method sometimes yields a result like Fig. 3.4-(c) (mainly due to super-pixels between two text-lines). In order to address this problem, the ideas in [50] is adopted: the minimum distance path in binary images is found based on dynamic programming and this separation is considered a new split proposal. This idea is

heuristic, however, note that these proposals are incorporated into the framework in a principled way: a new proposal is accepted only when it yields a lower energy than the current one.

3.2 Experimental Results of Various Handwritten Databases

In this section, experimental results of proposed method on various handwritten documents are presented. In order to prove the language-independent property of the proposed text-line segmentation algorithm, extensive experiments are performed on following databases:

- HIT-MW Database [27] → Chinese
- ICDAR 2009 Handwriting Segmentation Database [28]
  → Latin-based languages (English, French, German and Greek)
- ICDAR 2013 Handwriting Segmentation Database [29]
  → Latin-based and Bangla(Traditional Indian) languages
- IAM Handwriting Database [30] → English with some machine-printed part
- UMD Handwritten Arabic Database [31] → Arabic

Also, comparison of text-line segmentation results with the previous algorithm [11] is presented on each database.

3.2.1 Evaluation Measure

For the objective evaluation, the protocol of [28,29] is adopted. The MatchScore [51] is defined as

$$\text{MatchScore}(i,j) = \frac{|G_j \cap R_i|}{|G_j \cup R_i|}, \tag{3.28}$$

where R_i is a set of pixels labeled as the i-th text line by the algorithm, G_j is a set of pixels labeled as the j-th text-line by the ground truth, and $|\cdot|$ denotes the number of pixels in a set. When the MatchScore between detected text-lines and ground truth lines is greater than T, the pair is considered a one-to-one match. Detection rate and recognition accuracy are defined as

$$DR = \frac{o2o}{N}, RA = \frac{o2o}{M}, \tag{3.29}$$

where N is the number of text-lines in the ground truth, M is the number of text-lines in detected result, and $o2o$ is the number of one-to-one matches. The F-measure is defined as

$$F = \frac{2 \cdot DR \cdot RA}{DR + RA}, \tag{3.30}$$

and it is used as a performance measure.

3.2.2 Parameter Selection

In order to determine the optimal set of parameters (α, β) in (3.11) and (3.12), F-measure of ICDAR 2013 evaluation set is calculated for 42 sets of parameters. The set of parameters (α, β) is constructed with

$$\alpha = \{20.0, 25.0, 27.5, 30.0, 32.5, 35.0, 40.0\} \tag{3.31}$$

and

$$\beta = \{4.0, 4.5, 5.0, 5.5, 6.0, 6.5\}\,. \tag{3.32}$$

Table 3.1: Performance analysis of a set of parameters (α, β) on ICDAR 2013 Evaluation Set. Columns and rows represent different α and β respectively.

	20.0	25.0	27.5	30.0	32.5	35.0	40.0
4.0	98.18%	98.00%	98.39%	98.34%	98.28%	98.28%	98.11%
4.5	98.07%	98.50%	98.33%	98.24%	98.32%	98.28%	98.43%
5.0	98.56%	98.65%	98.64%	**98.71%**	98.66%	98.65%	98.32%
5.5	98.22%	98.13%	98.37%	98.47%	98.30%	98.37%	98.67%
6.0	98.47%	98.17%	98.30%	98.20%	98.24%	98.34%	98.64%
6.5	98.18%	98.00%	98.62%	98.47%	98.58%	98.45%	98.17%

Results of F-measure with various pairs of parameters are shown in Table 3.1. As shown, the variation of segmentation performance is not varying rapidly with a range of parameters (α, β). Based on this experiment, a pair that gives the best performance is selected where α in (3.11) is set to 30, and β in (3.12) is set to 5.0 in all experiments. For the determination of the number of seed points n in (3.8), it is set to 100 in order to achieve computation time reduction and accurate stroke width estimation. Finally, a threshold T that determines a pair is one-to-one matched is set to 0.95 in all databases as in the competitions [28, 29].

3.2.3 Experiment on HIT-MW Database

The HIT-MW database [27] is consisted of binary handwritten Chinese document images by multiple writers for off-line Chinese handwritten character recognition and

Table 3.2: Experimental results on the HIT-MW Database [27]. Numbers in parentheses are differences with the top method.

	DR	RA	F-Measure
Docstrum [1]	65.38% (34.40)	55.62% (44.26)	60.11% (39.72)
S. Tonghua *et al.* [52]	55.34% (44.34)	76.67% (23.08)	64.28% (35.43)
M. Arivazhagan *et al.* [41]	92.07% (7.71)	92.28% (7.60)	92.17% (7.66)
D. Xiaojun *et al.* [53]	95.92% (3.86)	96.86% (3.02)	96.39% (3.44)
F. Yin *et al.* [6]	98.03% (1.75)	97.53% (2.35)	97.78% (2.05)
H. Koo *et al.* [11]	99.68% (0.10)	99.75% (0.13)	99.71% (0.12)
Proposed	**99.78%**	**99.88%**	**99.83%**

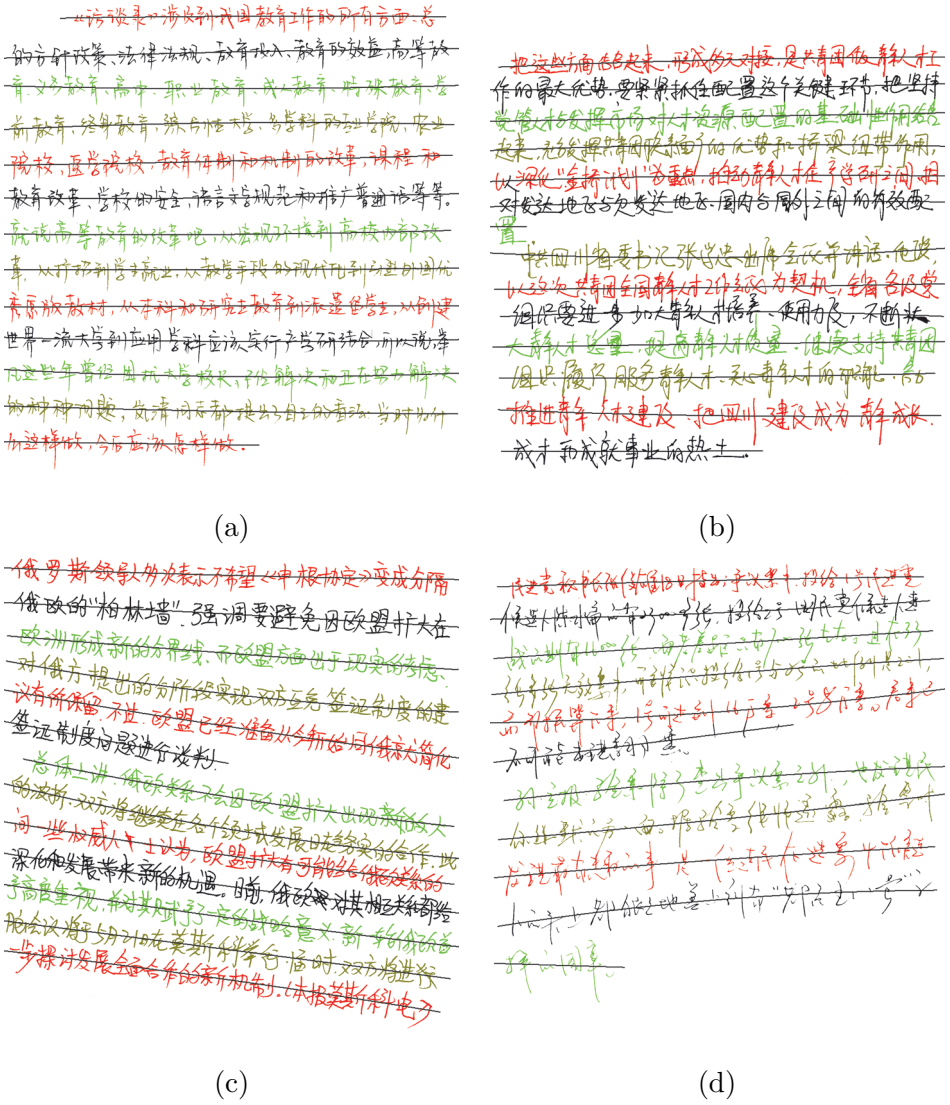


Figure 3.5: Text-line extraction results of proposed algorithm on HIT-MW database in various cases. (a) Straight text-lines. (b) Narrow spacings between the text-lines. (c) Slanted text-lines. (d) Different writing style.

text-line segmentation. In HIT-MW database, there are 853 scanned handwritten Chinese documents by more than 780 writers. The number of text-lines is $8,674$ and pixel-wise ground truth is made for all text-lines. All images are scanned and binarized to have only two class information: text(black pixels) and background(white pixels). Thus, preprocessing such as scan noise removal, binarization and text/non-text classification are not required for text-line segmentation of this database.

Experimental results on HIT-MW database are shown in Table 3.2. As can be seen, the proposed method yields the best performance on both databases and also shows better performance than previous state-of-the-art algorithm of HIT-MW database [11]. Some examples are shown in Fig. 3.5. The proposed method can handle the various cases of Chinese scripts like dense text-lines (Fig. 3.5-(b), slanted text-lines (Fig. 3.5-(c)) and different writing styles (Fig. 3.5-(d)) as well as in the case of straight text-lines (Fig. 3.5-(a)).

3.2.4 Experiment on ICDAR 2009/2013 Handwriting Segmentation Databases

The ICDAR 2009/2013 handwriting segmentation databases by N. Stamatopoulos *et al.* [28, 29] are constructed to record recent advances in off-line handwriting segmentation. The ICDAR 2009 handwriting segmentation database [28] is consisted of 200 documents written by Latin-based languages(English, French, German and Greek) from multiple writers. Total number of text-lines in 200 test images is $4,034$. Also, in ICDAR 2013 handwriting segmentation database, there are 150 handwritten document images for improved segmentation performance of more challenging handwritten documents. It includes 100 document images written in Greek and English and 50 images written in Bangla and the number of text-lines is $2,649$. The

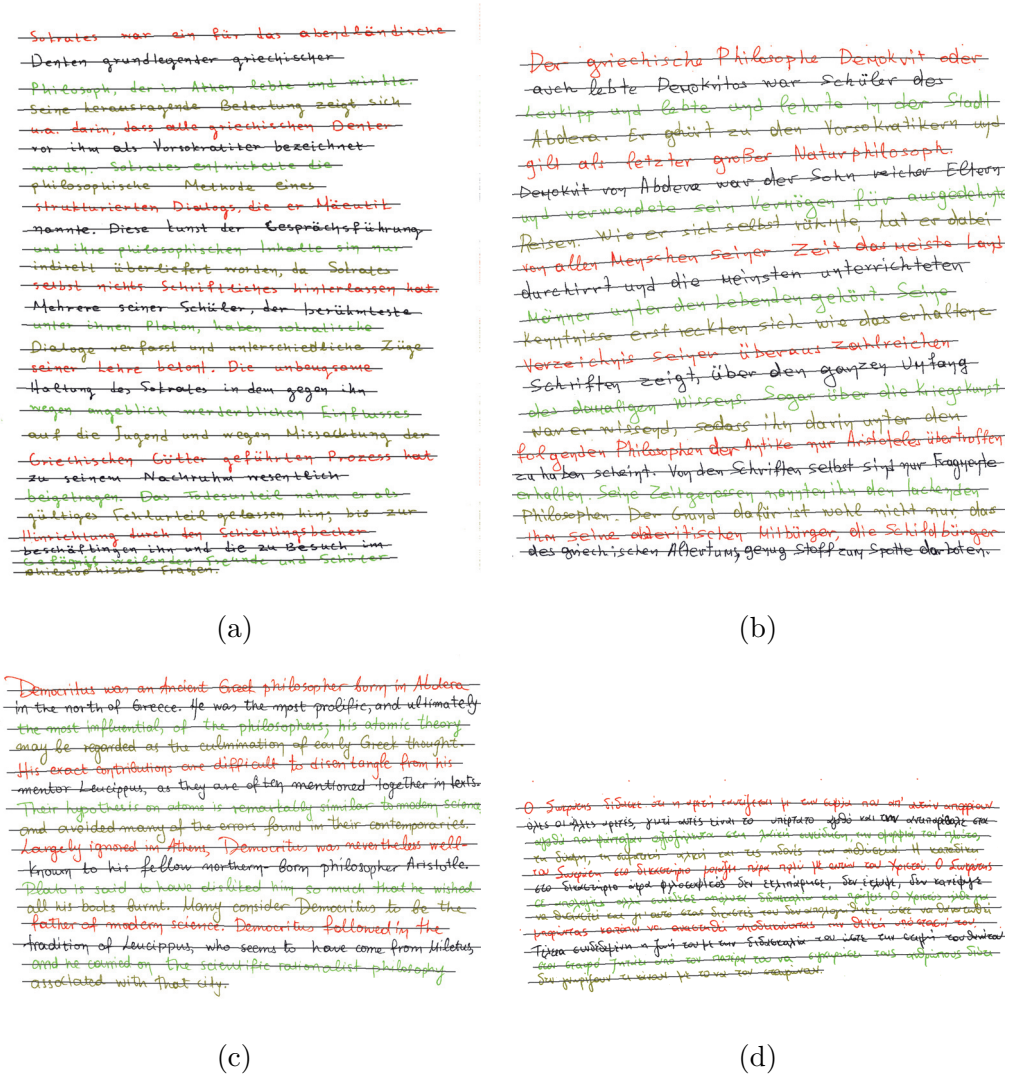


Figure 3.6: Text-line detection results of proposed algorithm on ICDAR 2009 handwriting segmentation contest testing database. (a) French script. (b) German script. (c) English script. (d) Greek script. Note that all documents are written in Latin-based languages and the structure of text-lines are relatively regular.

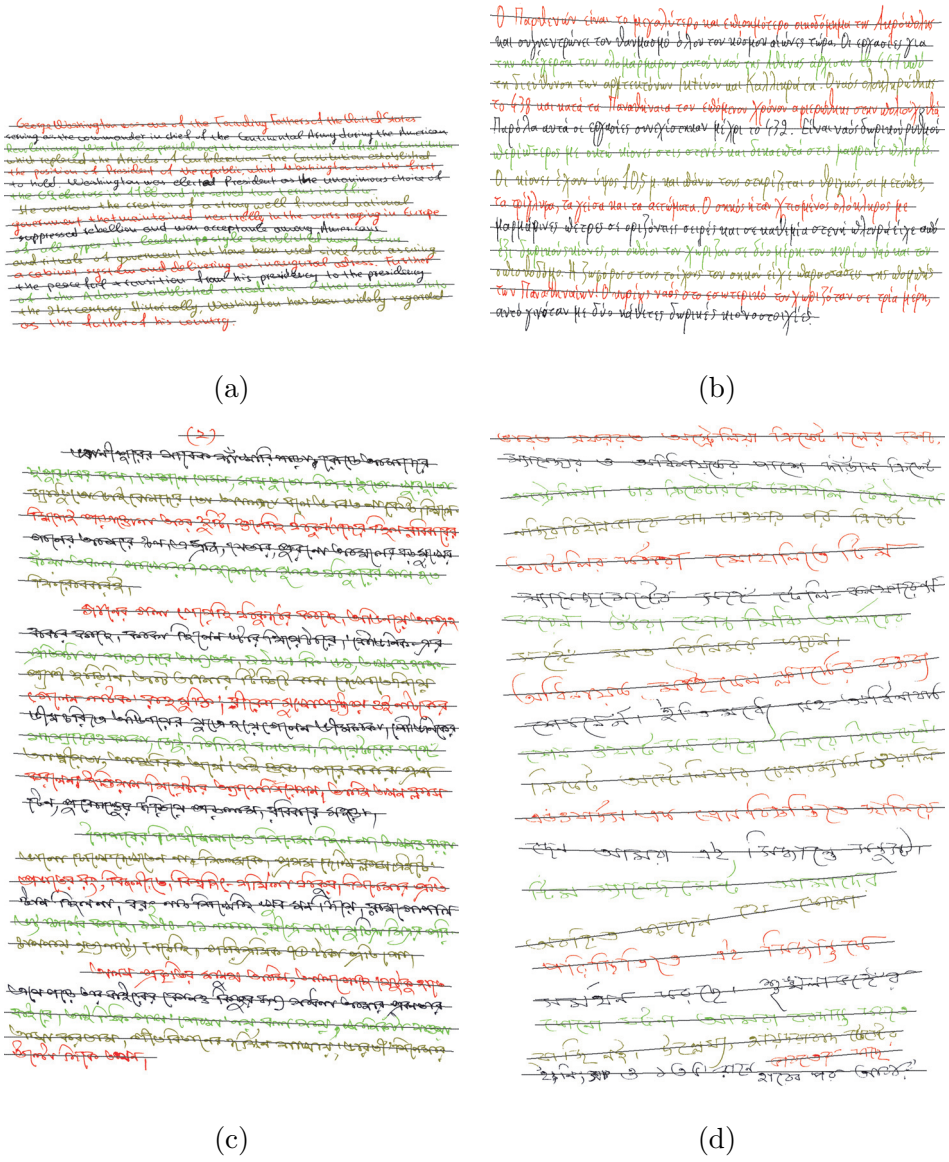


Figure 3.7: Some examples of the results of proposed algorithm on ICDAR 2013 handwriting segmentation contest testing database. (a) English script. (b) Greek script. (c), (d) Bangla script. In this set, the irregularity of the structure of text-lines is high and some characters written in Latin-based languages are touching across the lines due to long descenders.

images from ICDAR databases are scanned and binarized to have text(represented as black pixel) and background(represented as white pixel) pixels as in HIT-MW database.

Some examples of the text-line segmentation results are illustrated on Fig. 3.6 and Fig. 3.7. Clearly, the images of ICDAR 2013 database are more challenging than that of ICDAR 2009 database since the range of writing styles and languages of ICDAR 2013 database is much wider and the number of characters touching across between text-lines is larger than that of ICDAR 2009 database due to long descenders of characters. However, the proposed method can deal with those challenging cases since proposed super-pixel representation method enables correct estimation of spatially-varying states.

Experimental results of the proposed method, competition participating methods and some conventional methods on both databases are presented in Table 3.3 and Table 3.4, respectively. As shown, the proposed method records the best performance on ICDAR 2009 and ICDAR 2013 database among the previous researches as well as the participating methods. The proposed algorithm was submitted to ICDAR 2013 text-line segmentation competition, which ranked 1st in text-line detection among 10 other participants. Note that proposed method uses the same parameters for all experiments while some algorithms are based on machine learning and trained independently on HIT-MW and ICDAR 2009 database [6].

3.2.5 Experiment on IAM Handwriting Database

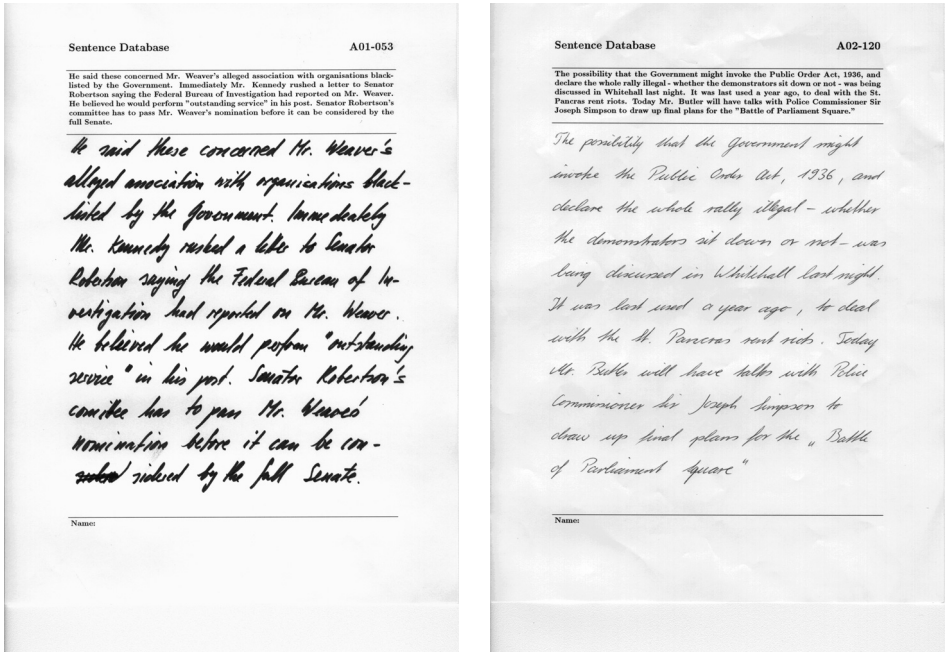
The IAM Handwriting Database [30] contains 1,539 pages of handwritten English script which have total 13,353 text-lines. It is constructed to evaluate text-line and word segmentation performance, text recognition, writer identification and verifica-

Table 3.3: Experimental results on the ICDAR 2009 Handwriting Segmentation Contest Evaluation Set [28]. Results of recent researches are also presented†. Numbers in parentheses are differences with the top method.

	DR	RA	F-Measure
Aegean [54]	77.59% (22.01)	77.21% (22.42)	77.40% (22.22)
Porto [55]	94.47% (5.13)	94.61% (5.02)	94.54% (5.08)
LRDE [56]	96.70% (2.9)	88.20% (11.43)	92.25% (7.37)
CUBS [57]	99.55% (0.05)	99.50% (0.13)	99.53% (0.09)
ILSP [15]	99.16% (0.44)	98.94% (0.69)	99.05% (0.57)
CASIA-MSTSeg [6]	95.86% (3.74)	95.51% (4.12)	95.68% (3.94)
CMM	98.54% (1.06)	98.29% (1.34)	98.42% (1.20)
ETS	86.66% (12.94)	86.68% (12.95)	86.67% (12.95)
Jadavpur Univ	87.78% (11.82)	86.90% (12.73)	87.34% (12.28)
PAIS	98.49% (1.11)	98.56% (1.07)	98.52% (1.10)
PPSL	94.00% (5.60)	92.85% (6.78)	93.42% (6.20)
REGIM	40.38% (59.22)	35.70% (63.93)	37.90% (61.72)
†M. Diem *et. al* [13]	98.59% (1.01)	98.59% (1.04)	98.59% (1.03)
†D. Fernández-Mota *et. al* [39]	98.40% (1.20)	95.00% (4.63)	96.67% (2.95)
†F. Yin *et al.* [6]	95.86% (3.74)	95.51% (4.12)	95.68% (3.94)
†H. Koo *et al.* [11]	98.31% (1.29)	98.05% (1.58)	98.18% (1.44)
Proposed	**99.60%**	**99.63%**	**99.62%**

Table 3.4: Experimental results on the ICDAR 2013 Handwriting Segmentation Contest Evaluation Set [29]. Results of conventional methods are also presented†. Numbers in parentheses are differences with the top method.

	DR	RA	F-Measure
CUBS [57]	97.96% (0.68)	96.94% (1.74)	97.45% (1.21)
GOLESTAN-a,b	98.23% (0.41)	98.34% (0.34)	98.28% (0.38)
LRDE	96.94% (1.67)	97.57% (1.11)	97.25% (1.41)
MSHK	91.66% (6.98)	90.06% (8.62)	90.85% (7.81)
NUS	98.34% (0.3)	98.49% (0.19)	98.41% (0.25)
QATAR-a	90.75% (7.89)	91.55% (7.13)	91.15% (7.51)
QATAR-b	91.73% (6.91)	93.14% (5.54)	92.43% (6.23)
CVC	91.28% (7.36)	89.06% (9.62)	90.16% (8.5)
IRISA [58]	97.85% (0.79)	96.93% (1.75)	97.39% (1.27)
†ILSP [15]	96.11% (2.53)	94.82% (3.86)	95.46% (3.2)
†NCSR [14]	92.37% (6.27)	92.48% (6.2)	92.43% (6.23)
†TEI [12]	97.77% (0.87)	96.82% (1.86)	97.30% (1.36)
†D. Fernández-Mota *et. al.* [39]	96.30% (2.34)	94.58% (4.10)	95.43% (3.23)
†H. Koo *et. al.* [11]	93.58% (5.06)	92.29% (6.39)	92.93% (5.73)
Proposed	**98.64%**	**98.68%**	**98.66%**



(a) (b)

Figure 3.8: Some forms of IAM Handwriting Database. The writing style and the stroke width of a document are different.

Table 3.5: Experimental results on the IAM database with all images. Numbers in parentheses are differences with the top method.

	DR	RA	F-Measure
H. Koo *et. al.* [11]	93.05% (5.64)	89.55% (8.95)	91.27% (7.32)
Proposed	**98.69%**	**98.50%**	**98.59%**

Table 3.6: Experimental results on the IAM database *Validation2* subset [30]. Some results are from [59]. Numbers in parentheses are differences with the top method.

	F-Measure
IUT [60]	37.70% (61.43)
CUBS [57]	96.70% (2.43)
TEI [12]	92.60% (6.53)
NCSR [14]	36.00% (63.13)
H. Koo *et. al.* [11]	95.20% (3.93)
Proposed Method	**99.13%**

tion experiments. All documents are scanned at a 300 dpi resolution and saved as a lossless compression (PNG format) with 256 grayscale level. Since the input of proposed segmentation algorithm is a binary document image, binarization with global thresholding and simple noise removal technique (remove scan noises) are applied. Some forms of IAM Handwriting Database are shown in Fig. 3.8. As shown, the documents of IAM database include machine-printed texts along with corresponding handwritten text scripts. Although the spaces between the text-lines are regular and clean, the writing style and stroke width of characters are diverse from documents.

In order to evaluate the performance on IAM database, experiments are conducted with whole images and its *Validation2* subset. For the experiment on whole set, F-measure of the proposed and previous method [11] is computed and the results are shown in Table 3.5. And for the *Validation2* subset, 4 other methods [12, 42, 57, 60] are also compared with the proposed method. The results are

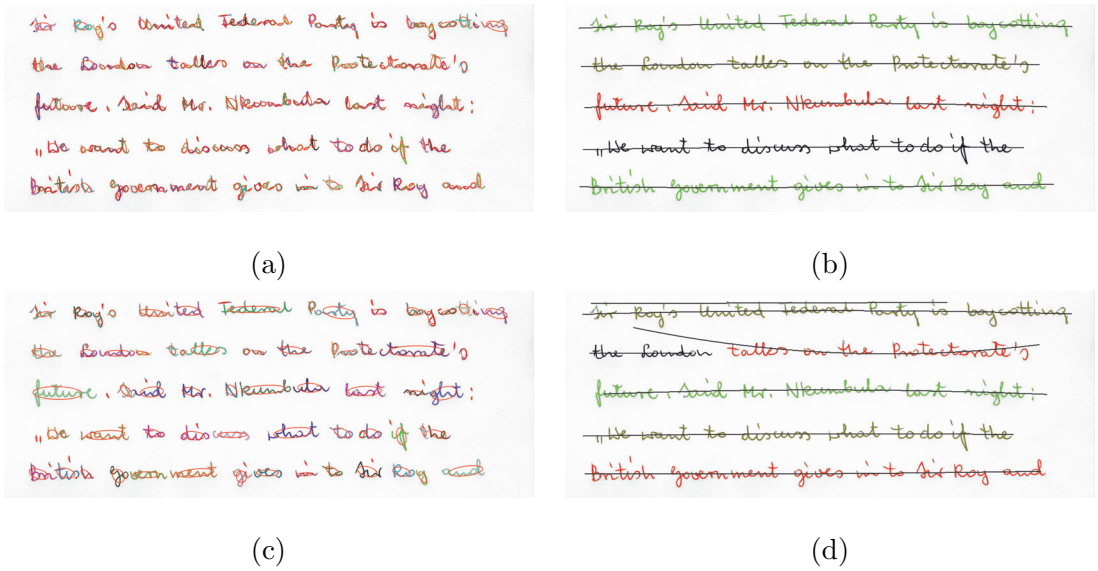
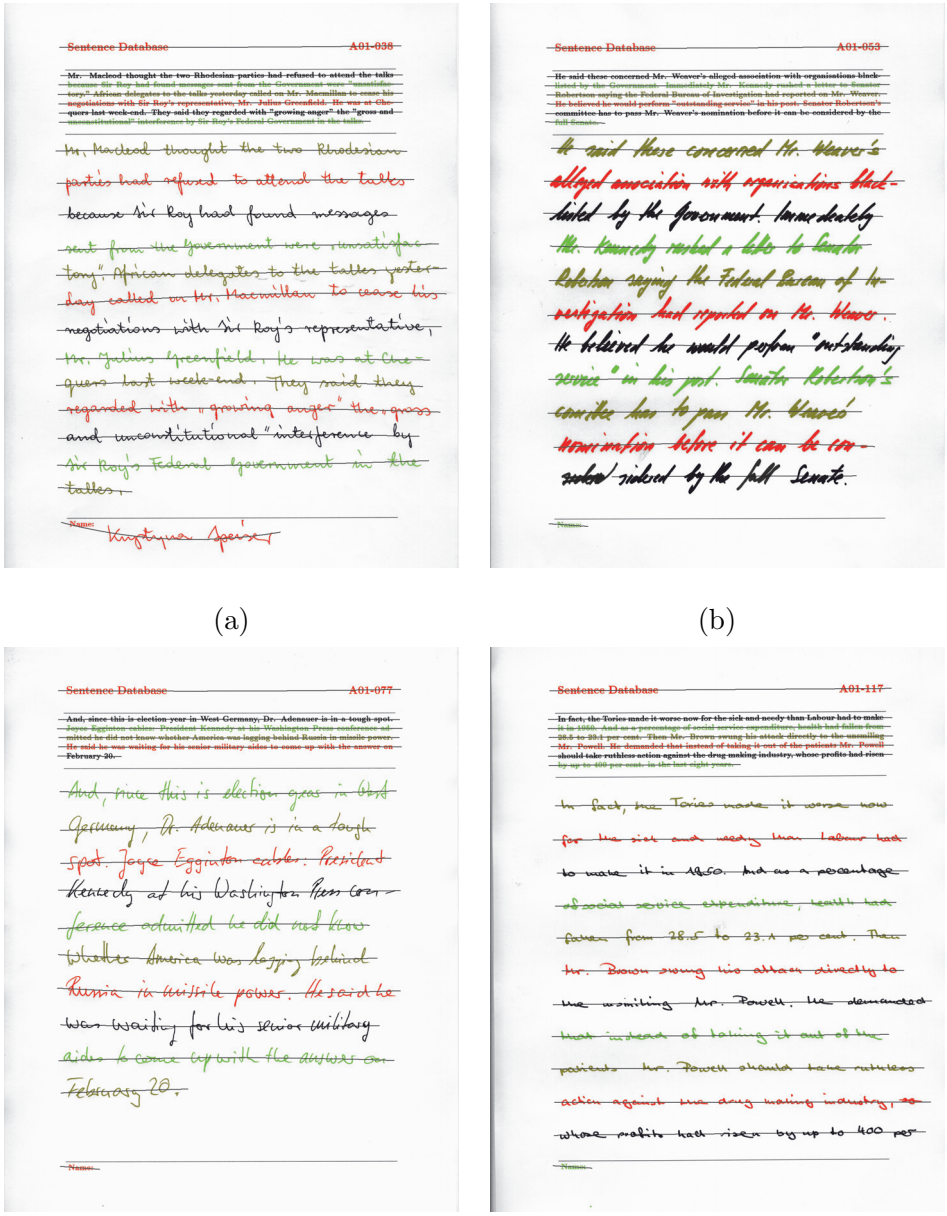


Figure 3.9: Comparison of text-line extraction results on IAM database. (a) Super-pixel representation of proposed method. (b) Segmentation result of the proposed method. (c) Super-pixel representation of [11].(d) Results of [11]. The previous method suffers from wrong text-line split due to tittles of some characters(the first row in (d)) and wrong text-line fitting by lack of spatially-varying state information due to cursive writing (the second row in (c) and (d)).



(a) (b)

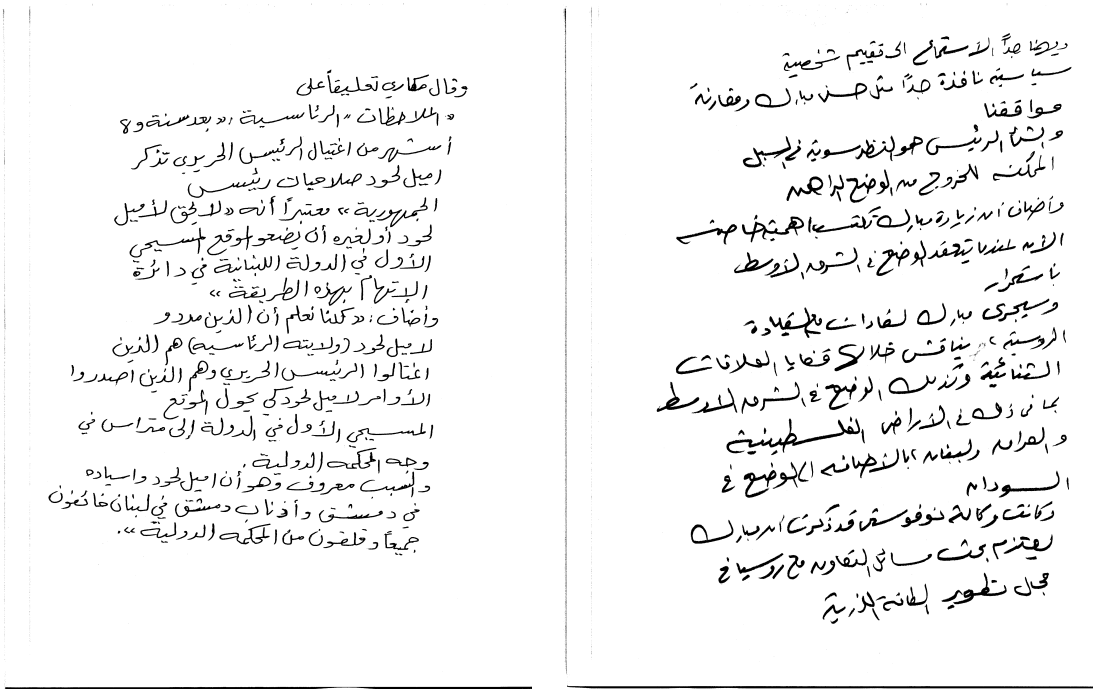
(c) (d)

Figure 3.10: Some examples of the text-line segmentation results of the proposed algorithm.

Table 3.7: Experimental results on the UMD database [31]. Numbers in parentheses are differences with the top method.

	DR	RA	F-Measure
J. Rodriguez-Serrano *et. al.* [61]	65.09% (30.14)	54.88% (1.86)	59.55% (35.75)
D. Fernández-Mota *et. al.* [39]	92.97% (2.26)	82.86% (12.52)	87.63% (7.67)
J. Kumar *et. al.* [62]	91.61% (3.62)	90.17% (5.21)	90.90% (4.40)
H. Koo *et. al.* [11]	82.76% (12.47)	77.81% (7.57)	80.21% (15.09)
Proposed Method	**95.23%**	**95.38%**	**95.30%**

presented in Table 3.6. The proposed method shows the best performance among previous methods in both whole set and *Validation2* subset. Comparison with the previous algorithm [11] is shown in Fig. 3.9. As illustrated, the proposed algorithm solves the problems of [11] which are induced by under-segmented super-pixels (due to cursive writing) and wrong text-line split due to tittles by exploiting the new super-pixel representation method and a new seam carving based proposal for energy minimization. Some results of the proposed method are shown in Fig. 3.10. The proposed algorithm can deal with the various stroke width (Fig. 3.10-(b)) and writing styles. Also, the proposed algorithm can deal with the scanned machine printed script without modifications since the assumptions of the energy equation (3.13 holds for the machine-printed documents as well as handwritten scripts.



(a) (b)

Figure 3.11: UMD Handwritten Arabic Database examples.

3.2.6 Experiment on UMD Handwritten Arabic Database

The UMD Handwritten Arabic Database [31] is constructed to evaluate the text-line detection and separation capabilities for Arabic handwritten documents. It includes 123 Arabic binarized document scanned with 5100 × 6600 resolution and provides pixel-wise labeled ground-truth with 1949 text-lines. The characteristic of Arabic script is similar to that of cursive writing of Latin-based languages as shown in Fig. 3.11: most characters constructing a word are connected to each other. Experimental results on UMD database are presented in Table 3.7. The proposed

(a) (b) (c) (d)

Figure 3.12: Text-line segmentation results of the proposed algorithm on UMD Arabic Handwritten database.

algorithm achieves F-measure 95.30% on 123 Arabic handwritten documents, which is the best performance among 5 algorithms with a significant difference. Some results of the proposed algorithm on UMD database are shown in Fig. 3.12. The proposed algorithm can also deal with the cases of slanted text-lines (Fig. 3.12-(c)) on Arabic handwritten documents.

In Fig. 3.13 and Fig. 3.14, text-line results of the proposed method and [11] are presented. Although the proposed method based on the previous energy minimization based algorithm [11], F-measure is increased about 15% from previous method. As can be seen in Fig. 3.13, a new *split* proposal based on the seam-carving successfully deal with the wrong text-line segmentation with some tittles of some characters like Latin-based cases. Also, the new super-pixel representation method makes states of super-pixels be correctly estimated thus gives a better text-line extraction results as in Fig. 3.14-(b).

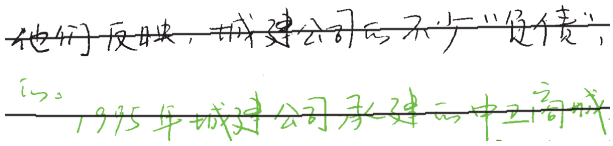
3.2.7 Limitations

Also, some failure cases of the proposed algorithm are illustrated in Fig. 3.15. As shown, proposed method suffers from the merge of adjacent text-lines when a text-line has a very small number of CCs and/or it is very close to its adjacent line. In these cases, the merge of adjacent lines introduces a negligible amount of fitting error increase and proposed method suffers from under-segmentation as shown in Fig. 3.15.

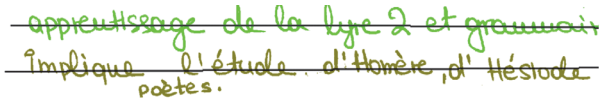
(a) (b) (c) (d)

Figure 3.13: Comparison of text-line extraction results with previous energy-based algorithm [11] on UMD database. (a), (b) Super-pixel representation/segmentation result of proposed method. (c), (d) Super-pixel representation/segmentation result of [11]. The previous method [11] suffers from wrong text-lines by tittles of some characters.

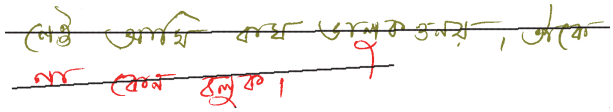
Figure 3.14: Another comparison of text-line extraction results with previous energy-based algorithm [11] on UMD database. (a), (b) Super-pixel representation/segmentation result of proposed method. (c), (d) Super-pixel representation/segmentation result of [11]. Due to the under-segmented super-pixels from [11], spatially-varying states of super-pixels are not correctly estimated which introduces wrong text-line results.



(a)



(b)



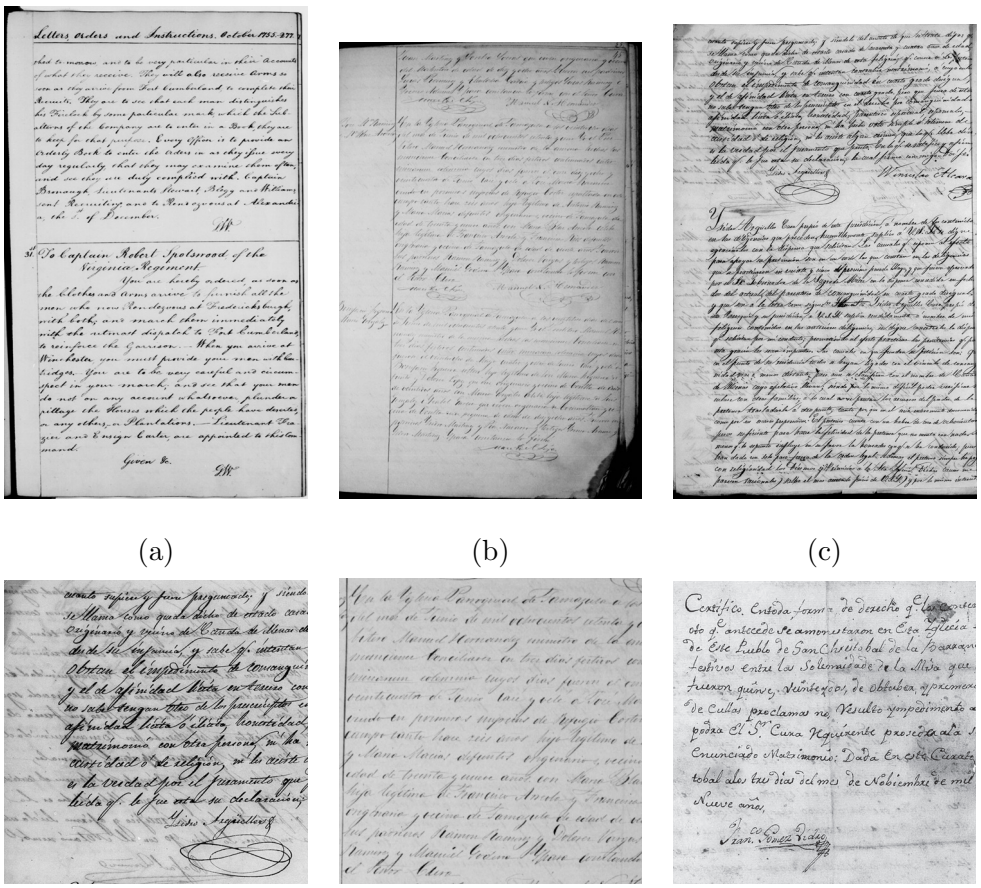
(c)

Figure 3.15: Failure cases. (a) Chinese script of HIW-MW dabtabase. (b) Latin-based script of ICDAR 2009 database. (c) Indian Script of ICDAR 2013 database.

Chapter 4

Preprocessing Method of Historical Document for Text-line Detection

In this chapter, an extended application of the text-line detection framework to historical documents is presented. The characteristics of text-lines in historical documents are similar to those of handwritten documents: (i) curvilinear text-lines, (ii) irregular neighborhood system of super-pixels. However, some preprocessing algorithms are needed to precede the text-line detection method since these types of documents suffer from various noise and degradations. Thus, in this chapter, binarization method for detecting text-lines of historical documents is presented. Then, experimental results on George Washington database [32] and ICDAR 2015 ANDAR dataset [33] are presented to evaluate the performance of the proposed method.



(a) (b) (c)

(d) (e) (f)

Figure 4.1: Example of the historical documents and degradations. (a),(b),(c) Historical documents from ICDAR 2015 ANDAR Dataset [33]. (d) Bleed-through from the other side of paper (e) Faint characters. (f) Stains.

4.1 Characteristics of Historical Documents

The text-line segmentation of historical documents is an essential task for preserving valuable historical manuscripts into digital data and also used for the layout

analysis and optical character recognition (OCR) which are necessary for processing historical documents. In general, contents of the historical documents are mainly handwritten manuscripts which can be handled by the proposed text-line segmentation algorithm. However, processing historical documents has some difficulties as illustrated in Fig. 4.1 unlike clean modern handwritten documents. They suffer from degradations such as bleed-through(Fig. 4.1-(d)), faint-characters(Fig. 4.1-(e)) and stains (Fig. 4.1-(f)). In order to extract text-lines in historical documents by text-line detection method, several preprocessing algorithms such as binarization for historical documents and noise removal are necessary.

Due to the difficulties as in Fig. 4.1, correct binarization with noise removal is important for the text-line segmentation and final OCR performance on historical documents. Thus, there have been a lot of researches, which can be classified into two main categories: global and local thresholding. In the case of global thresholding, a single threshold is exploited over the whole image to determine the foreground(text) and background(paper) pixels while local threshold method uses the threshold for each windows based on the characteristics of windows. There are some pioneering researches in this area: (i) Otsu's method [63] for global thresholding, (ii) Niblack's method [64], Sauvola's method [65] and Wolf's method [66] for local thresholding techniques. Otsu's method determines a threshold which minimizes within-class variance while maximizes inter-class variance. Since it provides good binarization result in a bimodal statistics like Fig. 4.1-(a) and requires relatively low complexity, some text-line segmentation algorithms [39] adopted it for its binarization step. However, due to its global thresholding nature, it cannot deal with the degradations such as stains, bleed-through and faint characters. On the other hand, the local thresholding method [64–66] calculates the threshold based on the statistics of

neighboring local pixels as they can detect characters effectively. Niblack method calculates the threshold $T = \mu + k\sigma$ for each pixel using the mean and variance (μ, σ) within the window however, it introduces a lot of background noise. Sauvola method and Wolf method suppress background noise from Niblack method but it also decreases recall rate of text.

4.2 A Combined Approach for the Binarization of Historical Documents

In order to take the advantages of both methods, a combined approach [67] is adopted with some improvements for the proposed text-line segmentation of historical documents. To prevent the binarization error from stains and faint-characters, they first estimate the background by simple inpainting method using the dilated Niblack binarization results as a mask. Then, normalization of image is followed to minimize the effect of variation of background and the normalized image is used as an input image to the combined framework of local and global binarization. For global binarization, Otsu binarization is performed on normalized image with post-processing that removes small components. However, it also removes faint characters which are not detected by Otsu method. In order to address this problem, Niblack binarization is adopted again to make a candidate for faint characters. The parameters of Niblack method are calculated from stroke width (window size) and image contrast (k). Finally, the binarization results of Otsu and Niblack method are integrated to produce the final result in a connected component(CC) level. For this, the connected components in Niblack result which have common foreground pixels in Otsu result are stored as a final result only if it is covered with sufficient amount

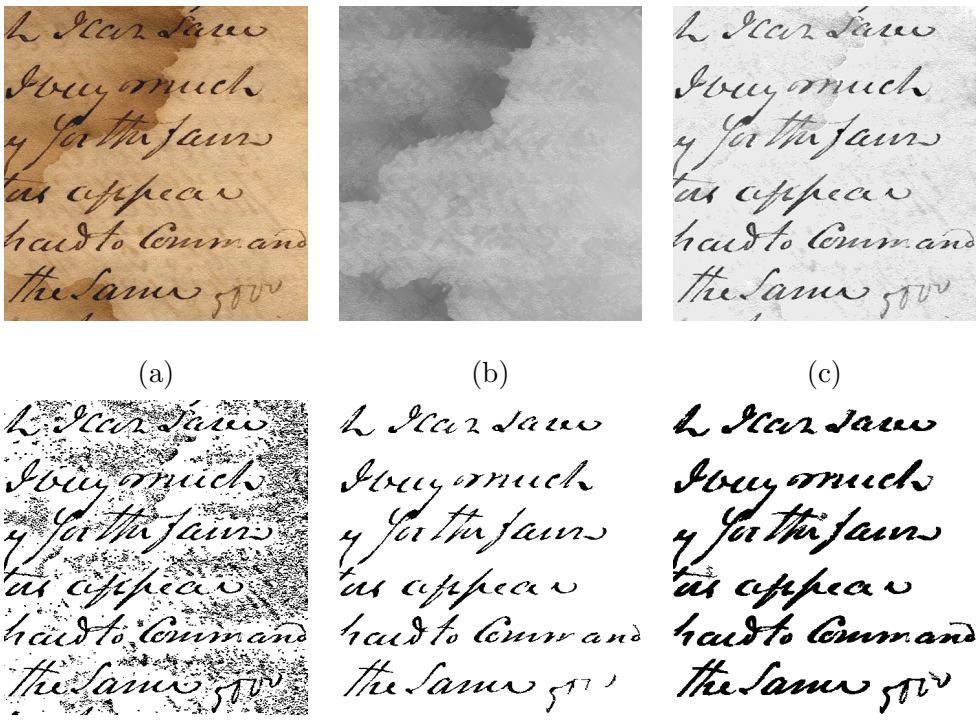


Figure 4.2: Example of images in historical image binarization method. (a) Original image. (b) Background estimation result. (c) Normalized image. (d) Results of Niblack method on (c). (e) Results of Otsu method on (c). (f) Final binarization result.

of Otsu output. The intermediate images from each step and results are illustrated in Fig. 4.2. According to their report, it achieves the best performance in 3 publicly available binarization contest databases (DIBCO' [68], HDIBCO'10 [69] and DIBCO'11 [70]).

However, some drawbacks are found when it is applied to detect text-lines on

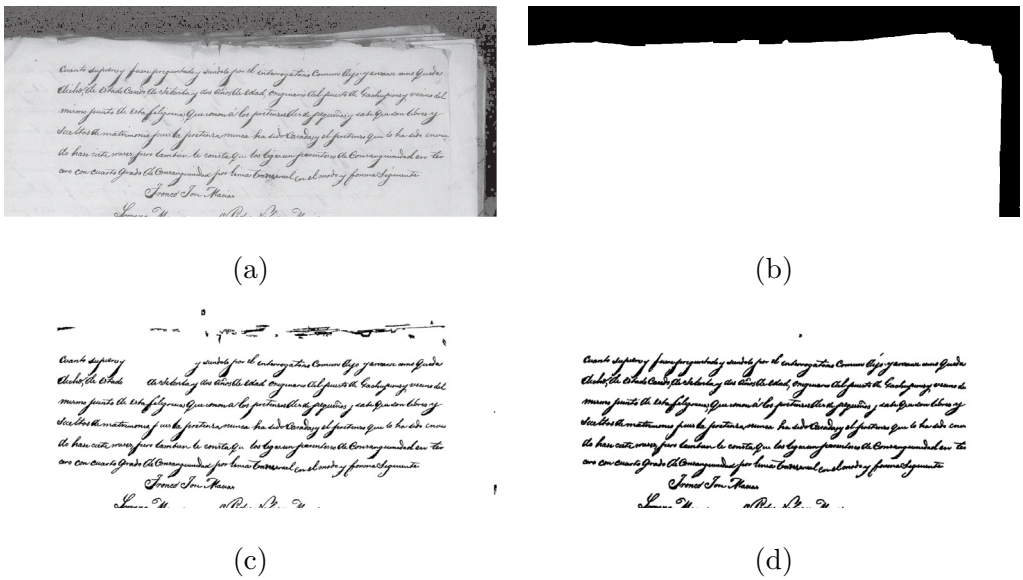


Figure 4.3: Effect of scanning boundary noise on binarization results. (a) Input image with boundary noise. (b) Proposed boundary estimation results. (c) Result of [67]. (d) Result with scan boundary rejection.

ICDAR 2015 database. It yields erroneous binarization results around the scanning boundary as in Fig. 4.3 since Niblack method has many false positives near the boundary. Also, binarization performance of fainted characters is significantly decreased when the difference between fainted character and background is very small as in Fig. 4.4. To address these problems, some modifications are applied to this binarization method. First, the scan boundary noise is eliminated by exploiting morphological operations. Usually, scan boundary noise exists near the edge of the image and their color is very dark as shown in Fig. 4.3-(a). To eliminate noises and characters in the image, erosion operation is applied to remove small connected components. Then, dilation operation with a large size $(41, 41)$ rectangular kernel is

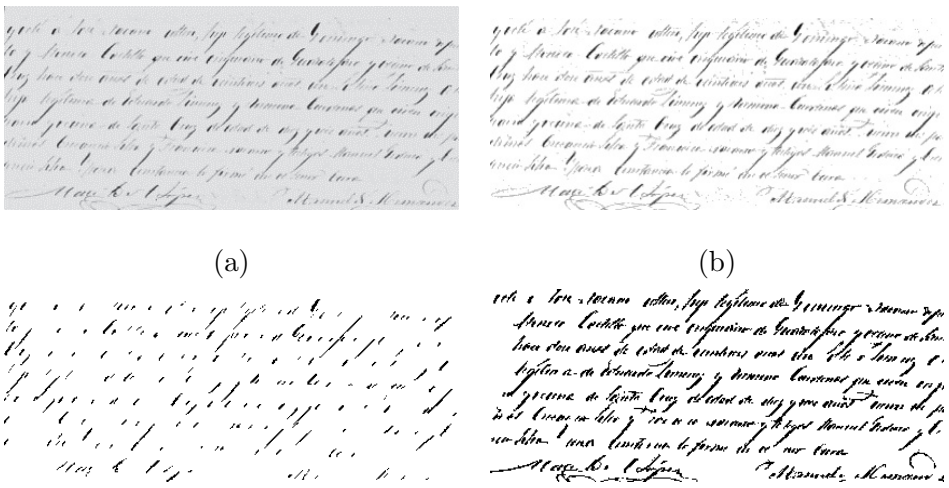


Figure 4.4: Effect of very faint character to binarization results. (a) Input image with faint character. (b) Proposed background forcing. (c) Binarization result of [67]. (d) Result of proposed method.

applied to detect the boundary noise region as illustrated in Fig. 4.3-(b). In binarization process, these regions are forced to be background so that the false positives in [67] are successfully rejected as shown in Fig. 4.3-(d).

Secondly, some modification is made to normalized image to alleviate the problem of detecting faint characters. As shown in Fig. 4.4-(a), the intensity difference between background and faint character is very small. In this case, both global and local threshold algorithm cannot distinguish the background and faint characters. Thus, pixels which have higher intensity than 0.9 is forced to have 1.0 intensity like in Fig. 4.4-(b). By applying this simple heuristic, better binarization results can be obtained as in Fig. 4.4-(d).

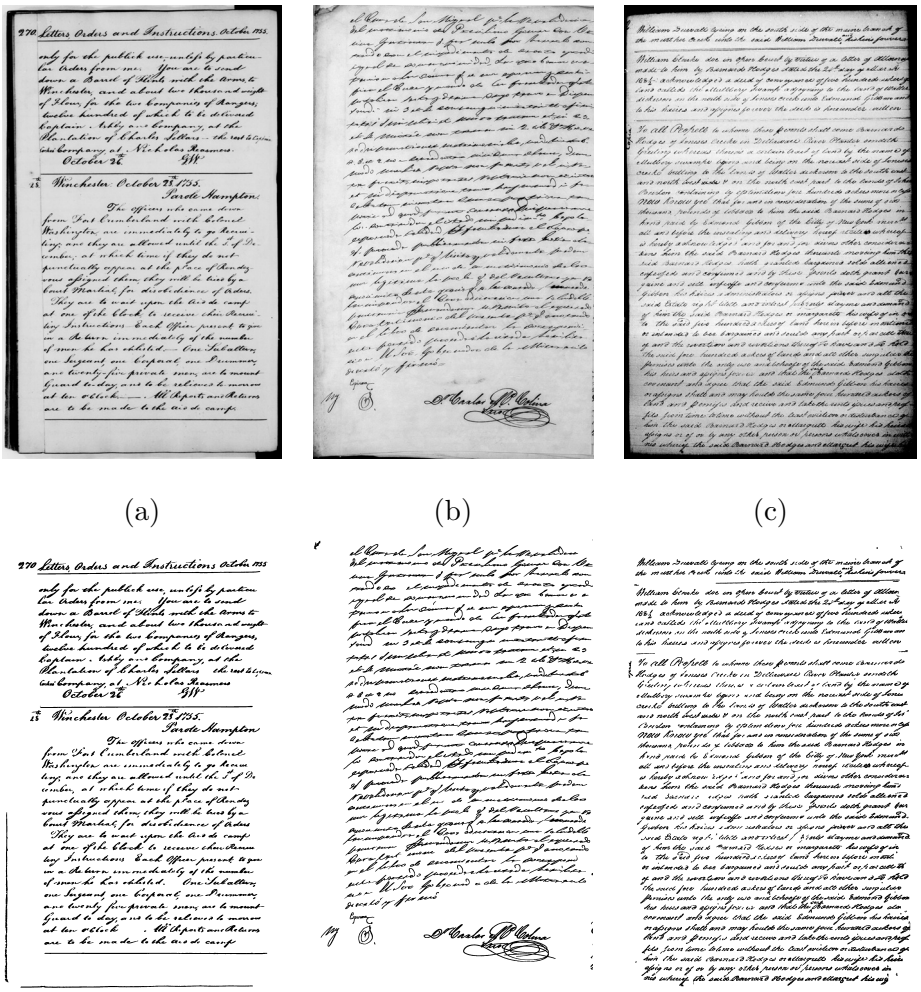


Figure 4.5: Some example of historical documents and its binarization results. (a) Document of George Washington manuscript. (b),(c) Documents of ICDAR 2015 database. (d) Binarization result of (a). (e) Binarization result of (b). (f) Binarization result of (c).

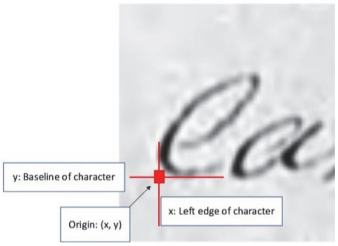


Figure 4.6: Illustration of OP (Origin Point) of ICDAR 2015 competition.

After applying binarization method to historical documents, it shows similar characteristics of clean handwritten documents as illustrated in Fig. 4.5. Thus, the proposed method can be applied to extract text-lines. However, there are some components such as horizontal lines, signatures and border noises which cannot be eliminated in binarization process. These components are removed in a text-line segmentation process by exploiting the state information (scales and orientations) of super-pixels.

4.3 Experimental Results of Text-line Detection for Historical Documents

4.3.1 Evaluation Measure and Configurations

For the evaluation of the performance of proposed binarization algorithm, the text-line detection algorithm is applied to two historical databases with proposed binarization method and 4 other methods. Then, the F-measure of text-line detection is compared as an objective measure. Since the text-line detection results of historical documents are dependent upon their binarization method, pixel-wise F-measure

cannot be exploited to evaluate the text-line detection performance of historical documents. Instead, Origin Point(OP)-based F-measure is adopted to measure the segmentation performance. The Origin Point (OP) coordinates are defined as the (x, y) coordinate located at the intersection between the baseline of first character of text-line and the left-most edge of that character as shown in Fig. 4.6. In George Washington database, the ground truth data are labeled with rectangular boxes. Thus, the Origin Points are manually annotated. In ICDAR 2015 database, the sequences of OP are given as a ground truth data.

The definition of OP-based F-measure is followed. Let $OP_y(i)$ is the y coordinate of i-th text-line and $EP_y(j)$ is the y coordinate of j-th text-line estimated by the algorithm. A pair of i-th text-line in ground truth and j-th text-line by algorithm is considered a *one-to-one* match if

$$|OP_y(i) - EP_y(j)| < \frac{|OP_y(i) - OP_y(i-1)|}{\sqrt{2}} \tag{4.1}$$

and

$$|OP_y(i) - EP_y(j)| < \frac{|OP_y(i) - OP_y(i+1)|}{\sqrt{2}}, \tag{4.2}$$

otherwise the pair is considered a miss. The detection rate, recognition accuracy and F-measure are defined as same as (3.29) like

$$DR = \frac{o2o}{N}, RA = \frac{o2o}{M}, F = \frac{2 \cdot DR \cdot RA}{DR + RA}, \tag{4.3}$$

where M is the number of text-lines by algorithm, N is the number of text-lines in ground truth and $o2o$ is the number of pairs that *one-to-one* matched.

Also, some modifications of text-line detection algorithm are made for the OP-based evaluations. Since the output of the proposed algorithm is pixel-wise labeled, it is needed to find a estimated origin point with the pixel-wise labeled results. For

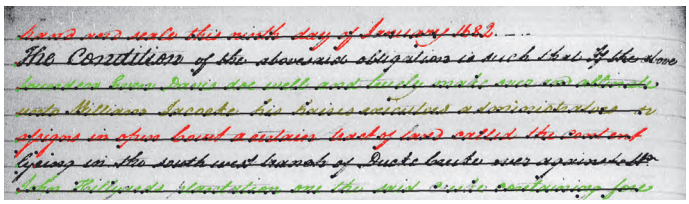


Figure 4.7: Illustration of estimating baseline of a text-line. Black dots are OPs of super-pixels and the black line is the estimated baseline.

this, a simple algorithm that finds a baseline of each text-line is conducted. First, a origin point of every super-pixel is set to the coordinate that have leftmost x and the lowest y . Since a lot of estimated OPs such as that of tittles, ascenders and descenders are not located at the baseline of text-lines, a RANSAC (RANdom SAmple Consensus) [71] with the first to second order polynomial is adopted to estimate the baseline of each text-line as shown in Fig. 4.7. For the estimated OPs of super-pixels of text-line, a leftmost inlier OP of the baseline is selected. Some adjustments on parameters of text-line extraction algorithm are applied to avoid over segmentation which is common in noisy historical documents. For this, sequence of parameters in bottom-up grouping is modified to $w_1 = 1.0, w_2 = 2.0, w_3 = 4.0, w_4 = 6.0, w_5 = 8.0$ while fixing $h = 0.1$ to prevent over segmentation of historical documents.

4.3.2 George Washington Database

The George Washington database [32] is consisted of 20 pages images from Library of Congress. All images are scanned at 300 dpi, 8 bit grayscale and the number of text-lines is 657. Experimental results including comparison with 4 binarization algorithms are shown in Table 4.1. As can be seen, proposed binarization method

Table 4.1: Experimental results on the George Washington database [32]. The F-measure is calculated by proposed text-line detection with corresponding binarization method. Numbers in parentheses are differences with the top method.

	DR	RA	F-Measure
Otsu *et. al.* [63]	97.26% (1.07)	99.22% (0.47)	98.23% (0.77)
Sauvola [65]	97.87% (0.46)	99.69%	98.54% (0.46)
Wolf [66]	97.41% (0.92)	99.38% (0.31)	98.39% (0.61)
K. Ntirogiannis *et. al.* [67]	97.41% (1.36)	99.69%	98.54% (0.46)
Proposed Method	**98.33%**	**99.69%**	**99.00%**

(a) (b) (c)

Figure 4.8: Text-line segmentation results with the proposed binarization method.

Table 4.2: Experimental results on the training set of ICDAR 2015 competition on historical text-line detection. [33]. The F-measure is calculated by proposed text-line detection with corresponding binarization method. Numbers in parentheses are differences with the top method.

	DR	RA	F-Measure
Otsu *et. al.* [63]	75.33% (10.15)	**85.85%**	80.24% (5.13)
Sauvola [65]	75.82% (9.66)	82.83% (3.02)	79.17% (6.20)
Wolf [66]	79.81% (5.67)	83.35% (2.50)	81.54% (3.83)
K. Ntirogiannis *et. al.* [67]	84.13% (1.35)	80.84% (5.01)	82.45% (2.92)
Proposed Method	**85.48%**	85.26% (0.59)	**85.37%**

shows the best performance on detecting text-lines of historical documents. Although images of George Washington database are historical manuscripts, they have small amount of noise and only a few characters are touching across the text-lines. Thus, text-line detection with proposed binarization method achieves similar text-line detection performance to that with conventional methods in George Washington database. Some examples of text-line segmentation results by proposed method are shown in Fig. 4.8.

4.3.3 ICDAR 2015 ANDAR Datasets

The ICDAR 2015 competition on text-line detection in historical documents is an ongoing competition for the evaluation of algorithms. For this competition, the organizer provides the training data set images with 726 scanned grayscale historical

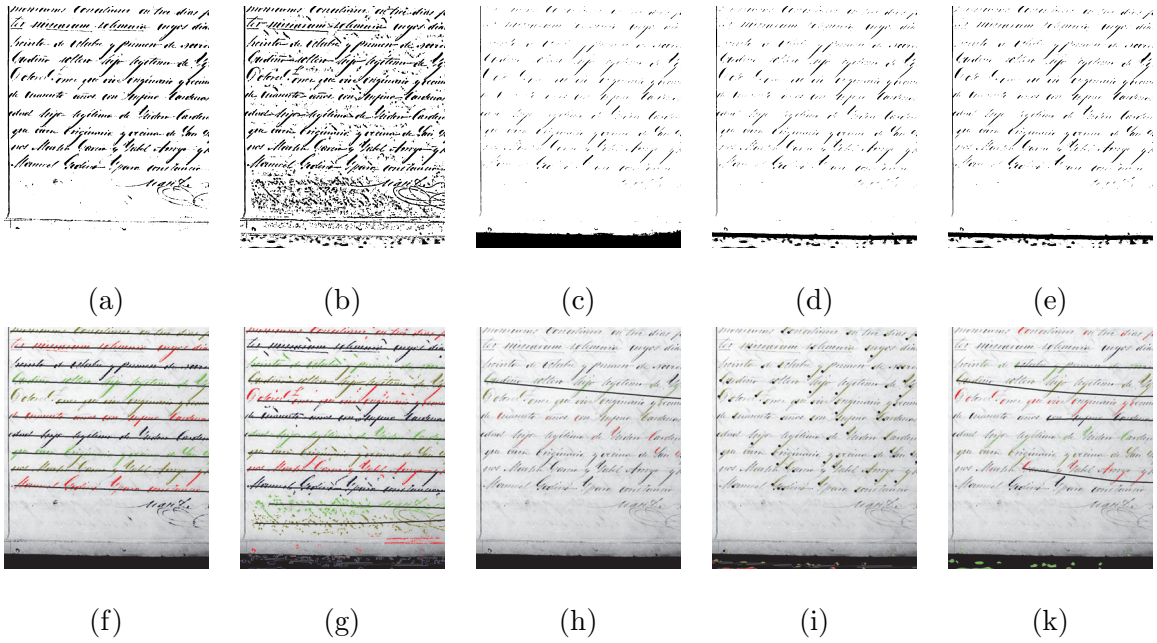


Figure 4.9: Comparison of proposed binarization method with 4 other methods on the part of historical documents. The first row represents binarization results and the second row represents text-line detection results. From left to right, proposed method, Ntirogiannis *et.al*, Otsu, Sauvola and Wolf methods.

documents in English, German and Spanish languages by multiple writers. The number of text-lines is 24, 188.

Experimental results on the training set (726 images) of ICDAR 2015 dataset with different binarization methods are shown in Table 4.2 and comparison is illustrated in Fig. 4.9. Since images of ICDAR 2015 database are very challenging and their degradation is severe, the text-line detection performance with conventional binarization methods is not good. However, proposed binarization method can successfully detect the characters in severely degraded historical documents thus

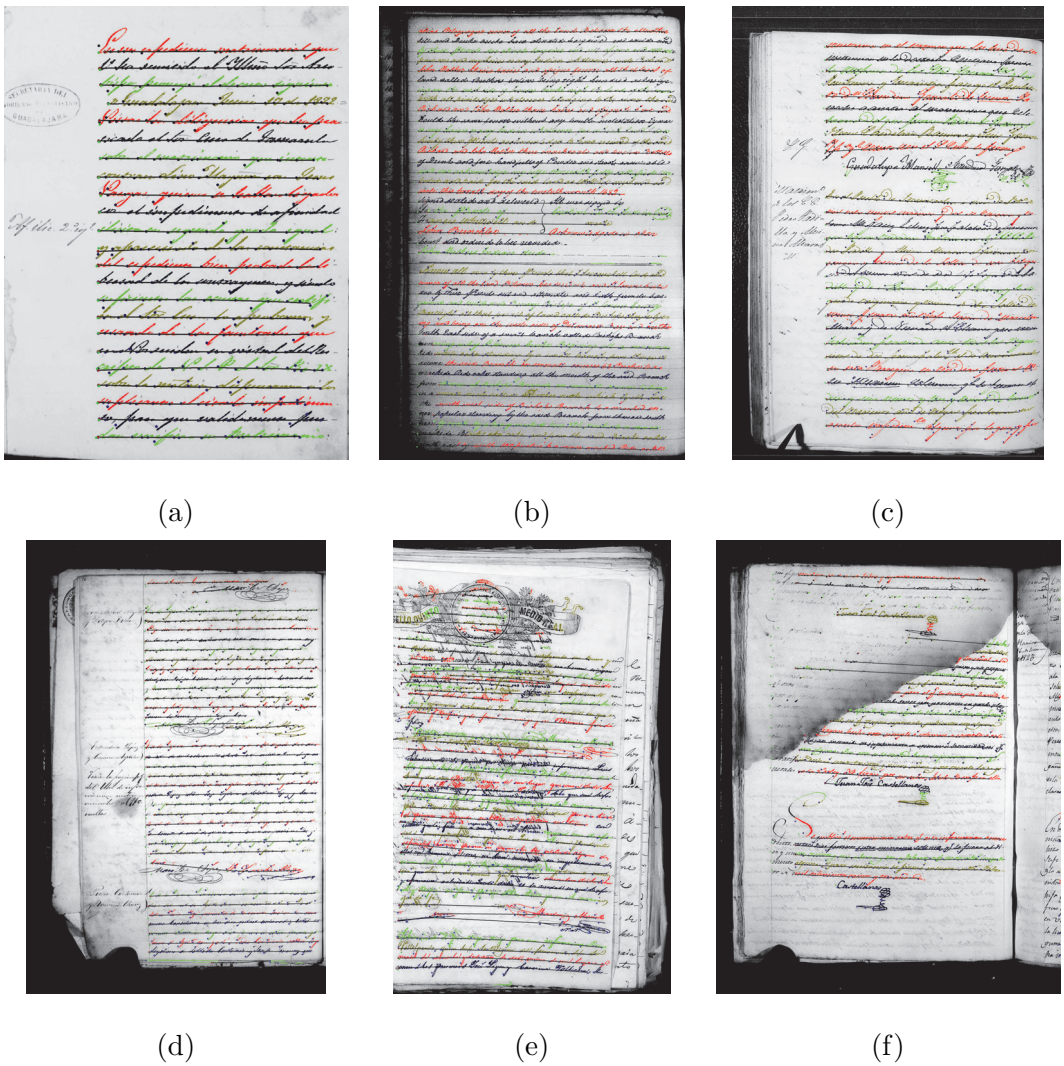


Figure 4.10: Text-line segmentation results of the proposed algorithm on ICDAR 2015 dataset and their F-measure. (a) 100 %. (b) 100 %. (c) 100 %. (d) 64.86%. (e) 33.8%. (f) 28.57 %

it yields the best text-line detection performance. For example, as shown in Fig. 4.9, proposed binarization method is able to detect faint characters which are not

detected by previous thresholding-based methods [63,65,66]. Also, it successfully removes the noise which are located around the scan boundary which are not rejected by previous method [67] by exploiting some heuristics.

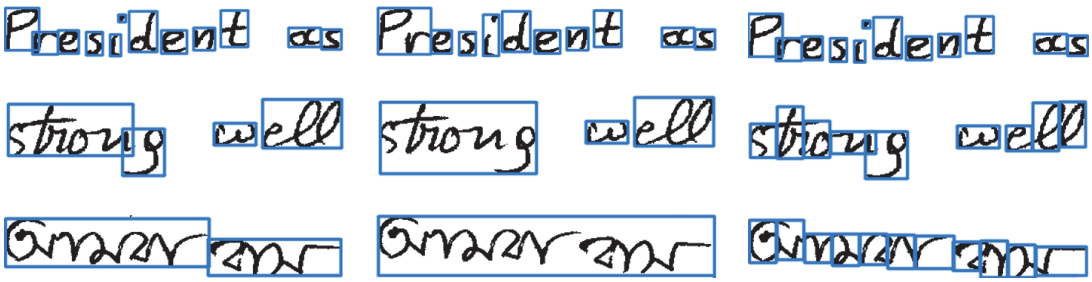
Some failure cases are shown in Fig. 4.10-(d)-(f). The proposed method cannot handle the case when non-textual handwritten components (signature, trade marks and scribbles) are not rejected (Fig. 4.10-(d)) and when the documents are severely degraded (Fig. 4.10-(e),(f)).

Chapter 5

Word Segmentation Method for Handwritten Documents based on Structured Learning

5.1 Proposed Approach for Word Segmentation

Word segmentation algorithm requires a text-line image which has already been segmented as in previous researches [14–19]. Like conventional methods, the word segmentation problem is considered a labeling problem that assigns a label (intra-word/inter-word gap) to each gap between characters in a given text-line in this dissertation. In this chapter, the normalized super-pixel representation method that extracts a set of candidate gaps in each text-line is presented. Then, a formulation of word segmentation problem as a binary quadratic problem is introduced, which allows to consider pairwise relations of gaps as well as local properties.



(a) (b) (c)

Figure 5.1: Illustration of super-pixel representation methods for different scripts and writing styles: (a) results of CC-based representation, (b) results of OC-based representation, (c) results of proposed representation method [34].

5.1.1 Text-line Segmentation and Super-pixel Representation

For the text-line segmentation, the algorithm in Chapter 3 is adopted which is ranked the first in ICDAR 2013 handwriting segmentation contest [29]. After the text-line extraction, components in each text-line is represented with proposed normalized super-pixels. In the literature, two approaches for the super-pixel representation are available: (i) connected components (CCs) based representation [15–17, 21, 24] and (ii) horizontally overlapping components (OCs) based representation [14,18,19]. However, super-pixel representation based on the above criteria may miss some candidates in the case of the cursive writings. Moreover, the input of this formulation is a set of super-pixels in a text-line, and the size and the number of super-pixels should be consistent across its script and/or the writing style. Therefore, super-pixels from conventional methods that focused on the detection of inter-word gaps are not appropriate for our formulation. That is, even though conventional methods successfully detect inter-word gaps as shown in Fig. 5.1-(a) and (b), the proposed

method prefers the representation in Fig. 5.1-(c) since the proposed method considers intra-word gaps as well as inter-word gaps. To this end, the idea of normalized CCs in [34] is adopted: the average stroke width $\overline{W}$ is estimated in a document and super-pixels by CC based representation are split ti several components so that they have normalized sizes (in terms of $\overline{W}$). With this method, the effect of written languages, contents, and scanning resolutions is significantly reduced in the super-pixel representation result as illustrated in Fig. 5.1.

5.1.2 Proposed Energy Function for Word Segmentation

Let us assume there are N gaps in a given text-line, where the i-th gap is denoted as x_i, and its label as $y_i \in \{0, 1\}$. Then, the word segmentation problem that assigns a binary label to each gap [14–19] can be posed as an energy optimization problem:

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} E(\mathbf{x}, \mathbf{y}) \tag{5.1}$$

where $\mathbf{x} = \{x_i\}_{i=1}^{N}$ and $\mathbf{y} = \{y_i\}_{i=1}^{N}$.

In the proposed method, the energy function $E(\mathbf{x}, \mathbf{y})$ is designed to reflect pairwise correlations as well as unary properties like that of text-line extraction. To be precise, in addition to unary terms (reflecting the individual likelihood of being either word-separator or not), two additional observations is encoded in this dissertation: (i) inter-word gaps should have similar features and (ii) the features of inter-word gap and intra-word gap should be different. Therefore the energy function is given by a pseudo-boolean function

$$\sum_{i} \left(a_i y_i + b_i(1 - y_i)\right) + \sum_{i<j} \left(c_{i,j} y_i y_j + d_{i,j}(y_i \oplus y_j)\right) \tag{5.2}$$

where $\oplus$ is XOR operator. In (5.2), a_i/b_i is the cost for x_i being either a word-separator or not, $c_{i,j}$ is the cost when both x_i and x_j are word-separators, and $d_{i,j}$ is

the cost when either x_i or x_j is a word-separator (the other is not a word-separator).

The energy function can be represented in a more compact way by assuming that the coefficients are linear functions of feature maps [36, 72]:

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} \langle \mathbf{w}, \Psi(\mathbf{x}, \mathbf{y}) \rangle \tag{5.3}$$

where

$$\Psi(\mathbf{x}, \mathbf{y}) = \left[\sum_i \psi_u(x_i) y_i, \sum_{i<j} \psi_p(x_i, x_j) y_i y_j, -\sum_{i<j} \psi_p(x_i, x_j)(y_i \oplus y_j) \right]. \tag{5.4}$$

Here, $\psi_u(x_i)$ is the unary feature of x_i, and $\psi_p(x_i, x_j)$ is pairwise feature. Since $\psi_p(x_i, x_j)$ reflects similarity between x_i and x_j, it becomes large as the properties of two gaps are similar, and vice versa.

The optimization of the function (5.3) is a binary quadratic assignment problem, which is an NP-hard problem [73]. However, since the number of gaps is usually small (e.g., $N < 100$), the optimization can be considered a Mixed-Integer Quadratic Programming problem (MIQP) [74]. An approximate solution can be obtained with well-developed techniques such as the branch-and-bound method [75].

5.2 Structured Learning Framework

For the word segmentation, the parameter $\mathbf{w}$ in (5.3) and feature maps should be determined. In this section, the feature map selection is discussed and adopted structured learning techniques is explained [76].

5.2.1 Feature Vector

In order to construct the feature map $\Psi(\mathbf{x}, \mathbf{y})$, the following features is adopted:

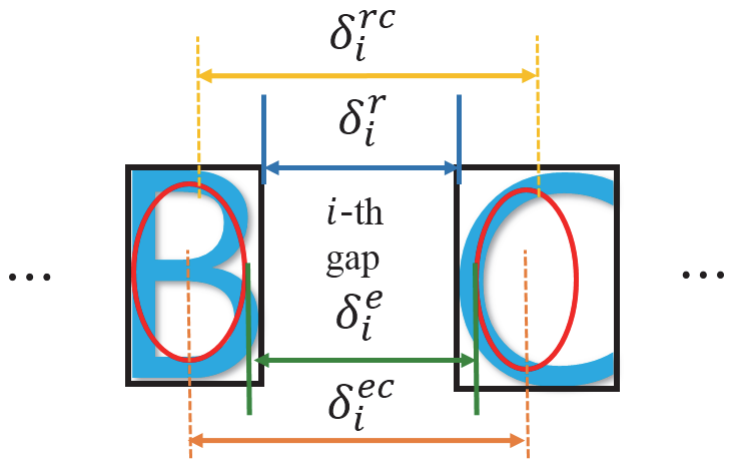


Figure 5.2: Illustration of the distance features. The super-pixels is represented with bounding boxes and ellipses [34], and four measures for gap-widths are employed.

Normalized distances between neighboring super-pixels ($\delta_i^r, \delta_i^{rc}, \delta_i^e, \delta_i^{ec}$)

The most salient property for word-separators is its large width (compared with intra-word gaps), and conventional methods already exploited this feature [16, 18, 19]. This property is also adopted in proposed algorithm, however, 4 measures to represent the width of gaps is exploited As shown in Fig. 5.2, they are boundary distances between rectangles/ellipses (denoted as δ_i^r and δ_i^e) and center-to-center distances of them (denoted as δ_i^{rc} and δ_i^{ec}). In order to achieve invariance to the capturing resolution, these distances are normalized with the estimated mean stroke width $\overline{W}$.

Projection profile features ($p_i^n, n = 1, \ldots, 5$)

The projection profile of a text-line is one-dimensional array that shows the number of pixels for each horizontal position. Thus, the zero-run (the length of consecu-

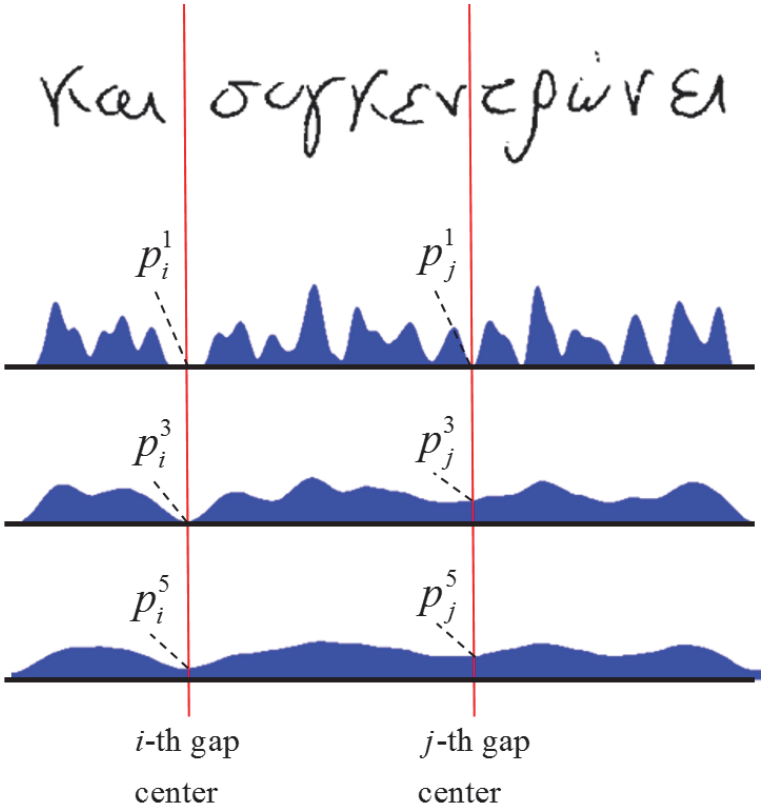


Figure 5.3: Illustration of the smoothed projection profile features. The first row is a part of handwritten text-line and the other rows are Gaussian filtered projection profiles with different kernel sizes ($\overline{W}, 3\overline{W}$, and $5\overline{W}$ respectively.)

tive zeros) of projection profile has been exploited for the word segmentation of machine-printed documents [4]. However, in handwritten documents, zero-run features become less salient because letters in different words are likely to touch each other and the skew (or curve) of a text-line may corrupt the zero-run in the projection profile. In order to address these difficulties, multiple Gaussian filters to projection profiles is applied, where the kernel sizes are set to be proportional to the stroke width $\overline{W}$ (in order to achieve the invariance to scales, e.g., scanning res-

olution). Then, the value of filtered projection profile at the gap center is used as a feature, which is denoted as p_i^n $(n = 1, 2, \ldots, 5)$ respectively as illustrated in Fig. 5.3. Also, they are normalized with $\overline{W}$ for scale invariant property.

Width ratio between current gap and the largest gap in a text-line (r_i)

Since the width ratio between a word-separator and the largest gap:

$$r_i = \frac{\delta_i^r}{\max_i \delta_i^r}. \tag{5.5}$$

is likely to be much larger than that of intra-word gaps. The ratio is also employed as a feature for word segmentation.

Based on above features, 10-dimensional unary feature is defined $\psi_u(x_i)$ as

$$\psi_u(x_i) = [\delta_i^r, \delta_i^e, \delta_i^{rc}, \delta_i^{ec}, p_i^1, p_i^2, p_i^3, p_i^4, p_i^5, r_i] \in \Re^{10}. \tag{5.6}$$

For the pairwise feature map (that should reflect the similarity between two gaps), a element-wise squared difference is adopted as in [73]:

$$\psi_p(x_i, x_j) = -|\psi_u(x_i) - \psi_u(x_j)|^2 \in \Re^{10}. \tag{5.7}$$

Thus, the dimension of the proposed feature map $\Psi(\mathbf{x}, \mathbf{y})$ in (5.4) is 30.

5.2.2 Parameter Estimation by Structured SVM

For the optimization of energy function (5.3), the parameter $\mathbf{w}$ is found with a structured learning technique. Given M training samples $\{(\mathbf{x}^n, \mathbf{y}^n)\}_{n=1}^M$ (M text-lines), n-slack Structured SVM [36] is formulated to estimate the optimal $\mathbf{w}$:

$$\min_{\mathbf{w},\zeta} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^{M} \zeta_n, \tag{5.8}$$

Algorithm 1 Cutting Plane Algorithm [76]

Input: M training samples $\{(\mathbf{x}^n, \mathbf{y}^n)\}_{n=1}^{M}, C, \epsilon$

$\mathcal{S}_n \leftarrow \phi, \zeta_n \leftarrow 0, \forall n$

repeat

  for $n = 1$ to M **do**

    $\hat{\mathbf{y}} \leftarrow \arg\max_{\mathbf{y}} \left(\Delta(\mathbf{y}^n, \mathbf{y}) + \langle \mathbf{w}, \Psi(\mathbf{x}^n, \mathbf{y}) \rangle\right)$

    if $\Delta(\mathbf{y}^n, \hat{\mathbf{y}}) - \langle \mathbf{w}, \Psi(\mathbf{x}^n, \mathbf{y}^n) - \Psi(\mathbf{x}^n, \hat{\mathbf{y}}) \rangle > \zeta_n + \epsilon$ **then**

      $\mathcal{S}_n \leftarrow \mathcal{S}_n \cup \{\hat{\mathbf{y}}\}$

    end if

  end for

  $(\mathbf{w}^*, \zeta^*) \leftarrow \arg\min_{\mathbf{w}, \zeta'} \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{n=1}^{M} \zeta'_n$

  subject to

  $\langle \mathbf{w}, \Psi(\mathbf{x}^n, \mathbf{y}^n) - \Psi(\mathbf{x}^n, \mathbf{y}') \rangle \geq \Delta(\mathbf{y}^n, \mathbf{y}') - \zeta'_n,$

  $\mathbf{y}' \in \mathcal{S}_n, \zeta'_n \geq 0, \forall n$

  $\mathbf{w} \leftarrow \mathbf{w}^*, \zeta \leftarrow \zeta^*$

until no $\mathcal{S}_n$ has changed during iteration

subject to

$$\begin{aligned} \langle \mathbf{w}, \Psi(\mathbf{x}^n, \mathbf{y}^n) - \Psi(\mathbf{x}^n, \mathbf{y}) \rangle \geq \Delta(\mathbf{y}^n, \mathbf{y}) - \zeta_n, \forall n \\ \zeta_n \geq 0, \forall n. \end{aligned} \tag{5.9}$$

For the loss function $\Delta(\mathbf{y}^n, \mathbf{y})$, Hamming distance is adopted defined as

$$\Delta(\mathbf{y}^n, \mathbf{y}) = \sum_i (y_i^n + y_i - 2y_i^n y_i). \tag{5.10}$$

Since the quadratic minimization problem (5.8) has an exponentially large number of constraints (5.9), the cutting plane algorithm [76] is adopted to solve (5.8) efficiently: the algorithm finds the most violated constraints by optimizing

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} \left(\Delta(\mathbf{y}^n, \mathbf{y}) + \langle \mathbf{w}, \Psi(\mathbf{x}^n, \mathbf{y}) \rangle \right). \tag{5.11}$$

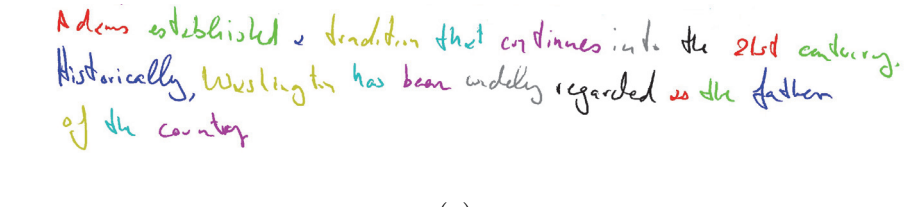
and tries to find the optimal $\mathbf{w}$. Note that the sub-problem that finds the most violated constraints can also be formulated as the binary quadratic assignment problem, since the Hamming loss function can be decomposed into the feature map [77]. The cutting plane algorithm is summarized in Algorithm 1.

5.3 Experimental Results

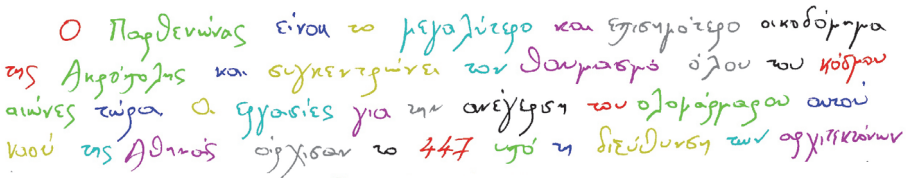
For the evaluation of the proposed word segmentation algorithm, experiments on two publicly available database is conducted: ICDAR 2009 and ICDAR 2013 handwriting segmentation contest databases [28, 29]. ICDAR 2009 database is composed of 100 training images and 200 test images. Even though they were written by various writers, all images are Latin-based scripts. On the other hand, ICDAR 2013 database (consisting of 200 training images and 150 test images) seems to be a more challenging set as it includes Latin-based and Indian scripts. For the training of

Table 5.1: Experimental results on the ICDAR 2009 and ICDAR 2013 Handwriting Segmentation Contest Evaluation Set [28, 29]. Some results are from [28, 29]. Numbers in parentheses are differences with the top method. †: 1st in 2013 competition, ‡: 1st in 2009 competition.

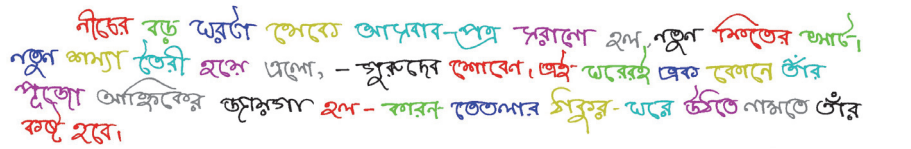
	ICDAR 2013 [29]			ICDAR 2009 [28]			Average
	DR	RA	F-Measure	DR	RA	F-Measure	
GOLESTAN-a†	89.66% (0.84)	90.44% (1.11)	90.05% (0.98)	N/A	N/A	N/A	N/A
GOLESTAN-b	89.59% (0.91)	90.07% (1.48)	89.83% (1.20)	N/A	N/A	N/A	N/A
NCSR [14]	88.31% (2.19)	90.98% (0.57)	89.62% (1.41)	N/A	N/A	N/A	N/A
PAIS	N/A	N/A	N/A	91.83% (3.63)	89.29% (5.41)	90.54% (4.54)	N/A
ILSP [16]‡	87.93% (2.57)	88.37% (3.18)	88.15% (2.88)	95.16% (0.30)	94.38% (0.32)	94.77% (0.31)	91.46% (1.60)
LRDE	86.75% (3.75)	86.94% (4.61)	86.84% (4.19)	88.56% (6.90)	79.74% (14.96)	83.92% (11.16)	85.38% (7.68)
CUBS [57]	87.86% (2.64)	86.91% (4.64)	87.38% (3.65)	89.62% (5.84)	84.45% (84.17)	86.96% (8.12)	87.17% (5.89)
Proposed	**90.50%**	**91.55%**	**91.03%**	**95.46%**	**94.70%**	**95.08%**	**93.06%**



(a)



(b)



(c)

Figure 5.4: Examples of proposed word segmentation results. (a) English script. (b) Greek script. (c) Traditional Indian(Bangla) script.

Structured SVM [36], each training set is used and the parameter C in (5.9) is set to 0.1. In order to solve binary quadratic assignment problems (5.3), (5.11), the MIQP solver by ILOG CPLEX [78] is exploited.

For the objective evaluation, the measures are adopted from [28,29] as in text-line extraction algorithm. To be precise, the MatchScore [51] is defined as

$$\text{MatchScore}(i,j) = \frac{|G_j \cap R_i|}{|G_j \cup R_i|}, \tag{5.12}$$

where G_j and R_i are two sets of pixels labeled as the i-th word by the algorithm and the j-th word by ground truth respectively, and $|\cdot|$ denotes the number of pixels in a set. The pair of a set of labeled pixels(by ground truth/algorithm) is considered

Table 5.2: Performance analysis with a different set of features on ICDAR 2009/2013 database. (Unit in F-measure.)

Database	2013 set	2009 set
With all features	91.03%	95.02%
w/o normalized distances (δ_i)	88.80%	94.68%
w/o projection profiles (p_i^n)	87.03%	88.04%
w/o width ratio (r_i)	90.01%	94.33%

a one-to-one match when the MatchScore is higher than 0.9. A detection rate (DR) and recognition accuracy is defined as

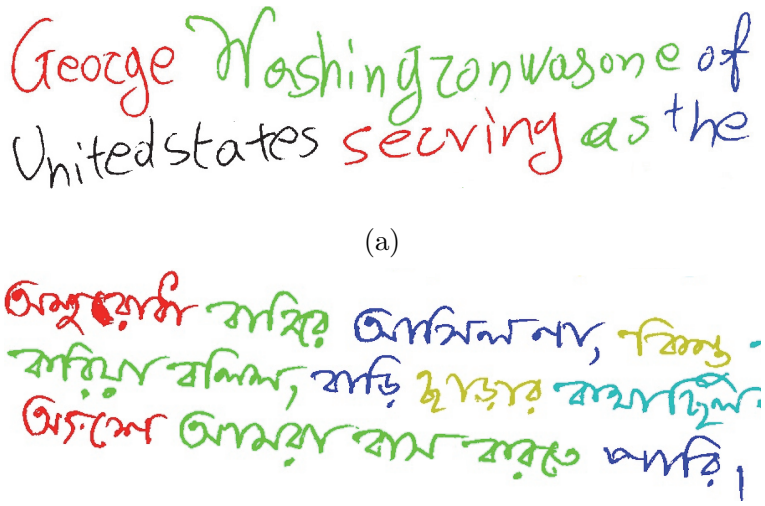
$$DR = \frac{o2o}{N}, RA = \frac{o2o}{M}, \tag{5.13}$$

where $o2o$ is the number of one-to-one matches, N is the number of words in the ground truth, and M is the number from the proposed algorithm. The performance metric F-measure (FM) is defined as a harmonic mean of DR and RA

$$FM = \frac{2 \cdot DR \cdot RA}{DR + RA}. \tag{5.14}$$

Experimental results on both databases are summarized in Table 5.1. As shown in the table, proposed method yields the largest F-measure both on ICDAR 2013 and 2009 databases. Some examples are illustrated in Fig. 5.4. As shown, the proposed structured-learning based framework works well for a variety of inputs. Since our method exploits various geometric properties and their weights are determined by

the Structured SVM method, it shows good segmentation performances on both sets which includes Latin-based and Indian documents from various writers. Also, an additional experiment by using a partial set of features is conducted for the evaluation of contribution of each feature. The results are shown in Table 5.2. As shown, the performance decreases from 0.28% (without r_i, 2009 database) to 6.98% (without p_i^n). That is, the results show that all the features have their discrimination powers. Some failure cases are illustrated in Fig. 5.5. As illustrated, the proposed method does not work when intra/inter word gaps have similar properties. The proposed word segmentation method takes $1 \sim 2$ seconds to process a handwritten document having $10 \sim 20$ text-lines with an Intel Core i5 PC with our un-optimized C++ implementation. Including the text-line extraction algorithm in Chapter 3, the total processing time is about $3 \sim 4$ seconds with Intel Core i5 CPU @ 2.8GHz. Some examples are shown in Fig. 5.4.



(a)

(b)

Figure 5.5: Failure cases. (a) English script. (b) Traditional Indian(Bangla) script.

Chapter 6

Conclusions

In this dissertation, the algorithms for the segmentation of document images into text-line and words have been proposed based on energy minimization framework with a super-pixel representation with normalized CC. The proposed segmentation algorithms are capable of extracting text-lines and words of handwritten documents with various languages and writers as well as machine-printed documents since the proposed algorithms extract regularized super-pixel for text component regardless of its contents.

Based on this super-pixel representations and their states, an energy equation whose optimization result yields text-lines has been developed in the first chapter of this dissertation. Further, the optimization technique for energy minimization framework has been improved to deal with the various documents with different languages. An extensive experiments have been conducted on various databases and the results show that the proposed algorithm yields the state-of-the-art performance while working in a language independent manner.

Then, this dissertation have proposed preprocessing algorithm for text-line de-

tection of historical documents. Unlike modern handwritten documents, historical documents suffer from various degradations which deteriorate the text-line detection performance. To address this problem, a combinational approach of global/local thresholding binarization is adopted and improved for the text-line detection of degraded historical documents. With proposed method, the text-lines of historical documents have been successfully detected.

Also, a word segmentation algorithm have been proposed in this dissertation as a second extension of text-line detection framework. Unlike conventional approaches, the proposed word segmentation problem has been formulated as a binary quadratic programming and the parameters have been estimated with the structured learning method. With the proposed formulation, the pairwise similarities between word-separators as well as unary properties can be taken into account in the word segmentation. Also, by using the Structured SVM, all parameters are estimated in a principled way and it is believed the proposed method can be easily extended to other databases. Experimental results on the databases consisted of Latin-based and Indian documents have shown that the proposed word segmentation algorithm yields the state-of-the-art performances.

Bibliography

[1] L. O'Gorman, "The document spectrum for page layout analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1162–1173, Nov. 1993.

[2] K. Kise, A. Sato, and M. Iwata, "Segmentation of page images using the area voronoi diagram," *Computer Vision and Image Understanding*, vol. 70, no. 3, pp. 370–382, Jun. 1998.

[3] H. I. Koo and N. I. Cho, "State estimation in a document image and its application in text block identification and text line extraction," in *European Conference on Computer Vision (ECCV)*, 2010, pp. 421–434.

[4] S. H. Kim, C. B. Jeong, H. K. Kwag, and C. Y. Suen, "Word segmentation of printed text lines based on gap clustering and special symbol detection," in *Proc. of Int. Conf. on Pattern Recognition*, vol. 2, 2002, pp. 320–323.

[5] Y. Y. Tang, S.-W. Lee, and C. Y. Suen, "Automatic document processing: a survey," *Pattern recognition*, vol. 29, no. 12, pp. 1931–1952, 1996.

[6] F. Yin and C.-L. Liu, "Handwritten Chinese text line segmentation by clustering with distance metric learning," *Pattern Recognition*, vol. 42, no. 12, pp. 3146–3157, Dec. 2009.

[7] G. Louloudis, B. Gatos, I. Pratikakis, and C. Halatsis, "Text line detection in handwritten documents," *Pattern Recognition*, vol. 41, no. 12, pp. 3758–3772, Dec. 2008.

[8] S. Bukhari, F. Shafait, and T. Breuel, "Script-independent handwritten textlines segmentation using active contours," in *International Conference on Document Analysis and Recognition (ICDAR)*, 2009, pp. 446–450.

[9] V. Bosch, A. H. Toselli, and E. Vidal, "Statistical text line analysis in handwritten documents," in *International Conference on Frontiers in Handwriting Recognition (ICFHR)*, 2011, pp. 201–206.

[10] A. Alaei, P. Nagabhushan, and U. Pal, "A new text-line alignment approach based on piece-wise painting algorithm for handwritten documents," in *International Conference on Document Analysis and Recognition (ICDAR) 2011*, 2011, pp. 324–328.

[11] H. I. Koo and N. I. Cho, "Text-line extraction in handwritten Chinese documents based on an energy minimization framework," *IEEE Trans. Image Process.*, vol. 21, no. 3, pp. 1169–75, Mar. 2012.

[12] A. Nicolaou and B. Gatos, "Handwritten text line segmentation by shredding text into its lines," in *Proc. of Int. Conf. on Document Analysis and Recognition (ICDAR)*, 2009, pp. 626–630.

[13] M. Diem, F. Kleber, and R. Sablatnig, "Text line detection for heterogeneous documents," in *Proc. of Int. Conf. on Document Analysis and Recognition (ICDAR)*, Aug. 2013, pp. 743–747.

[14] G. Louloudis, B. Gatos, I. Pratikakis, and C. Halatsis, "Text line and word segmentation of handwritten documents," *Pattern Recognition*, vol. 42, no. 12, pp. 3169 – 3183, Dec. 2009.

[15] T. Stafylakis, V. Papavassiliou, V. Katsouros, and G. Carayannis, "Robust text-line and word segmentation for handwritten documents images," in *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2008, pp. 3393–3396.

[16] V. Papavassiliou, T. Stafylakis, V. Katsouros, and G. Carayannis, "Handwritten document image segmentation into text lines and words," *Pattern Recognition*, vol. 43, no. 1, pp. 369 – 377, Jan. 2010.

[17] G. Seni and E. Cohen, "External word segmentation of off-line handwritten text lines," *Pattern Recognition*, vol. 27, no. 1, pp. 41–52, Jan. 1994.

[18] T. Varga and H. Bunke, "Tree structure for word extraction from handwritten text lines," in *Proc. of Int. Conf. on Document Analysis and Recognition (ICDAR)*, Aug 2005, pp. 352–356.

[19] S. H. Kim, S. Jeong, G. S. Lee, and C. Y. Suen, "Word segmentation in handwritten korean text lines based on gap clustering techniques," in *Proc. of Int. Conf. on Document Analysis and Recognition (ICDAR)*, 2001, pp. 189–193.

[20] R. Bozinovic and S. Srihari, "Off-line cursive script word recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 1, pp. 68–83, Jan 1989.

[21] U. Mahadevan and R. Nagabushnam, "Gap metrics for word separation in handwritten lines," in *Proc. of Int. Conf. on Document Analysis and Recognition (ICDAR)*, 1995, pp. 124–127.

[22] R. Manmatha and J. L. Rothfeder, "A scale space approach for automatically segmenting words from historical handwritten documents," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1212–1225, 2005.

[23] G. Kim, V. Govindaraju, and S. Srihari, "A segmentation and recognition strategy for handwritten phrases," in *Proc. of Int. Conf. on Pattern Recognition*, 1996, pp. 510–514.

[24] S. Srihari, H. Srinivasan, P. Babu, and C. Bhole, "Handwritten Arabic word spotting using the cedarabic document analysis system," in *Proc. Symposium on Document Image Understanding Technology*, 2005, pp. 123–132.

[25] Y. Y. Tang, S.-W. Lee, and C. Y. Suen, "Automatic document processing: A survey," *Pattern Recognition*, vol. 29, no. 12, pp. 1931 – 1952, 1996. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031320396000441

[26] H. I. Koo and D. H. Kim, "Scene text detection via connected component clustering and nontext filtering," *IEEE Trans. Image Process.*, vol. 22, no. 6, pp. 2296–2305, Jun. 2013.

[27] T. Su, T. Zhang, and D. Guan, "Corpus-based HIT-MW database for offline recognition of general-purpose Chinese handwritten text," *International Journal of Document Analysis and Recognition (IJDAR)*, vol. 10, no. 1, pp. 27–38, Jun. 2007.

[28] B. Gatos, N. Stamatopoulos, and G. Louloudis, "ICDAR 2009 handwriting segmentation contest," in *Proc. of Int. Conf. on Document Analysis and Recognition (ICDAR)*, 2009, pp. 1393–1397.

[29] N. Stamatopoulos, B. Gatos, G. Louloudis, U. Pal, and A. Alaei, "ICDAR 2013 handwriting segmentation contest," in *Proc. of Int. Conf. on Document Analysis and Recognition (ICDAR)*, 2013, pp. 1402–1406.

[30] U.-V. Marti and H. Bunke, "The IAM-database: an english sentence database for offline handwriting recognition," *International Journal on Document Analysis and Recognition*, vol. 5, no. 1, pp. 39–46, 2002.

[31] L. Kang, J. Kumar, P. Ye, and D. Doermann, "Learning text-line segmentation using codebooks and graph partitioning," in *Proc. of the Int. Conf. on Frontiers in Handwriting Recognition*. IEEE Computer Society, 2012, pp. 63–68.

[32] V. Lavrenko, T. M. Rath, and R. Manmatha, "Holistic word recognition for handwritten historical documents," in *Proc. of International Workshop on Document Image Analysis for Libraries*, 2004, pp. 278–287.

[33] "ICDAR 2015 ANDAR Competition on Text Line Detection in Historical Documents," http://http://collections.ancestry.com/ANDAR-TL-2015/Home.

[34] J. W. Ryu, H. I. Koo, and N. Cho, "Language-independent text-line extraction algorithm for handwritten documents," *IEEE Signal Process. Lett.*, vol. 21, no. 9, pp. 1115–1119, Sep. 2014.

[35] S. Kim, C. Jeong, H. Kwag, and C. Suen, "Word segmentation of printed text lines based on gap clustering and special symbol detection," in *Proc. of International Conference on Pattern Recognition*, vol. 2, 2002, pp. 320–323 vol.2.

[36] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," *Journal of Machine Learning Research*, pp. 1453–1484, Sep. 2005.

[37] L. Likforman-Sulem, A. Zahour, and B. Taconet, "Text line segmentation of historical documents: a survey," *International Journal of Document Analysis and Recognition (IJDAR)*, vol. 9, no. 2-4, pp. 123–138, Apr. 2006.

[38] B. Gatos, A. Antonacopoulos, and N. Stamatopoulos, "Handwriting segmentation contest," in *Proc. of Int. Conf. on Document Analysis and Recognition (ICDAR)*, vol. 2, 2007, pp. 1284–1288.

[39] D. Fernández-Mota, J. Lladós, and A. Fornés, "A graph-based approach for segmenting touching lines in historical handwritten documents," *International Journal on Document Analysis and Recognition (IJDAR)*, pp. 1–20, 2014.

[40] G. Nagy, S. Seth, and M. Viswanathan, "A prototype document image analysis system for technical journals," *Computer*, vol. 25, no. 7, pp. 10–22, Jul. 1992.

[41] M. Arivazhagan, H. Srinivasan, and S. Srihari, "A statistical approach to line segmentation in handwritten documents," in *in Proc. SPIE-Documet Recognition and Retrieval XIV*, vol. 6500, 2007, pp. 65 000T–1–65 000T–11.

[42] G. Louloudis, B. Gatos, I. Pratikakis, and C. Halatsis, "Text line and word segmentation of handwritten documents," *Pattern Recognition*, vol. 42, no. 12, pp. 3169–3183, Dec. 2009.

[43] L. Likforman-Sulem, A. Hanimyan, and C. Faure, "A hough based algorithm for extracting text lines in handwritten documents," in *Proc. of International*

Conference on Document Analysis and Recognition (ICDAR), vol. 2, Aug 1995, pp. 774–777 vol.2.

[44] Y. Li, Y. Zheng, D. Doermann, and S. Jaeger, "Script-independent text line segmentation in freestyle handwritten documents," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 8, pp. 1313–29, Aug. 2008.

[45] F. Wahlberg and A. Brun, "Graph based line segmentation on cluttered handwritten manuscripts," in *Proc. of International Conference on Pattern Recognition*, Nov 2012, pp. 1570–1573.

[46] M. d. Berg, O. Cheong, M. v. Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*, 3rd ed. Santa Clara, CA, USA: Springer-Verlag TELOS, 2008.

[47] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.

[48] R. Casey and E. Lecolinet, "A survey of methods and strategies in character segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 7, pp. 690–706, Jul. 1996.

[49] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2963–2970.

[50] R. Saabni and J. El-Sana, "Language-independent text lines extraction using seam carving," in *International Conference on Document Analysis and Recognition (ICDAR)*, 2011, pp. 563–568.

[51] I. Phillips and A. Chhabra, "Empirical performance evaluation of graphics recognition systems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 9, pp. 849–870, Sep. 1999.

[52] T. Su, T. Zhang, H. Huang, and Y. Zhou, "Skew detection for Chinese handwriting by horizontal stroke histogram," in *International Conference on Document Analysis and Recognition (ICDAR)*, vol. 2, 2007, pp. 899–903.

[53] X. Du, W. Pan, and T. D. Bui, "Text line segmentation in handwritten documents using Mumford-Shah model," *Pattern Recognition*, vol. 42, no. 12, pp. 3136 – 3145, Dec. 2009.

[54] E. Kavallieratou, N. Dromazou, N. Fakotakis, and G. Kokkinakis, "An integrated system for handwritten document image processing," *Int. J. Patt. Recogn. Artif. Intell.*, vol. 17, pp. 617–636, Jun. 2003.

[55] J. Cardoso, A. Capela, A. Rebelo, and C. Guedes, "A connected path approach for staff detection on a music score," in *IEEE International Conference on Image Processing (ICIP)*, 2008, pp. 1005–1008.

[56] T. Geraud. (2009) LRDE method for line segmentation. [Online]. Available: http://www.lrde.epita.fr/cgi-bin/twiki/view/Olena/ModuleIcdar

[57] Z. Shi, S. Setlur, and V. Govindaraju, "A steerable directional local profile technique for extraction of handwritten arabic text lines," in *International Conference on Document Analysis and Recognition (ICDAR)*, 2009, pp. 176–180.

[58] A. Lemaitre, J. Camillerapp, and B. Coasnon, "Handwritten text segmentation using blurred image," in *Proc. SPIE 9021, Document Recognition and Retrieval XXI*, vol. 9021, 2013, pp. 90 210D–90 210D–12.

[59] B. Moysset and C. Kermorvant, "On the evaluation of handwritten text line detection algorithms," in *Proc. of Int. Conf. on Document Analysis and Recognition (ICDAR)*. IEEE, 2013, pp. 185–189.

[60] A. Zahour, L. Likforman-Sulem, W. Boussalaa, and B. Taconet, "Text line segmentation of historical Arabic documents," in *Proc. of Int. Conf. on Document Analysis and Recognition (ICDAR)*, vol. 1. IEEE, 2007, pp. 138–142.

[61] J. A. Rodríguez-Serrano and F. Perronnin, "Handwritten word-spotting using hidden markov models and universal vocabularies," *Pattern Recognition*, vol. 42, no. 9, pp. 2106–2116, 2009.

[62] J. Kumar, W. Abd-Almageed, L. Kang, and D. Doermann, "Handwritten arabic text line segmentation using affinity propagation," in *Proc. of the 9th IAPR International Workshop on Document Analysis Systems*. ACM, 2010, pp. 135–142.

[63] "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. 9, no. 1, pp. 62–66, Jan 1979.

[64] W. Niblack, *An introduction to digital image processing*. Strandberg Publishing Company, 1985.

[65] J. Sauvola and M. Pietikäinen, "Adaptive document image binarization," *Pattern recognition*, vol. 33, no. 2, pp. 225–236, 2000.

[66] C. Wolf, J.-M. Jolion, and F. Chassaing, "Text localization, Enhancement and Binarization in Multimedia Documents," in *Proc. of the International Conference on Pattern Recognition*, vol. 2, 2002, pp. 1037–1040.

[67] K. Ntirogiannis, B. Gatos, and I. Pratikakis, "A combined approach for the binarization of handwritten document images," *Pattern Recognition Letters*, vol. 35, no. 0, pp. 3 – 15, 2014, frontiers in Handwriting Processing. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S016786551200311X

[68] B. Gatos, K. Ntirogiannis, and I. Pratikakis, "DIBCO 2009: document image binarization contest," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 14, no. 1, pp. 35–44, 2011. [Online]. Available: http://dx.doi.org/10.1007/s10032-010-0115-7

[69] I. Pratikakis, B. Gatos, and K. Ntirogiannis, "H-DIBCO 2010 - handwritten document image binarization competition," in *Proc. of International Conference on Frontiers in Handwriting Recognition (ICFHR)*, Nov 2010, pp. 727–732.

[70] ——, "ICDAR 2011 document image binarization contest (DIBCO 2011)," in *Proc. of International Conference on Document Analysis and Recognition (ICDAR)*, Sept 2011, pp. 1506–1510.

[71] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.

[72] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun, "Support vector machine learning for interdependent and structured output spaces," in *Proc. of the Int. Conf. on Machine Learning*, 2004, pp. 104–111.

[73] T. Caetano, J. McAuley, L. Cheng, Q. V. Le, and A. Smola, "Learning graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 6, pp. 1048–1058, Jun. 2009.

[74] R. Lazimy, "Mixed-integer quadratic programming," *Mathematical Programming*, vol. 22, no. 1, pp. 332–349, Jan. 1982.

[75] B. Borchers and J. E. Mitchell, "An improved branch and bound algorithm for mixed integer nonlinear programs," *Computers & Operations Research*, vol. 21, no. 4, pp. 359–367, Apr. 1994.

[76] T. Joachims, T. Finley, and C.-N. J. Yu, "Cutting-plane training of structural SVMs," *Machine Learning*, vol. 77, no. 1, pp. 27–59, Oct. 2009.

[77] S. Kim, S. Nowozin, P. Kohli, and C. Yoo, "Task-specific image partitioning," *IEEE Trans. Image Process.*, vol. 22, no. 2, pp. 488–500, Feb 2013.

[78] "IBM ILOG CPLEX Optimizer," http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/.

초록

문서 영상을 텍스트라인과 단어 단위로 분리하는 것은 광학 문자 인식(OCR), 문서 정렬, 레이아웃 분석 및 문서 영상 압축에 필요한 매우 중요한 과정이다. 따라서 인쇄 문서에 대한 텍스트라인 및 단어 분리에 대한 연구는 많이 진행되어 왔고, 해결된 문제로 볼 수 있다. 그러나 손글씨 문서에 경우는 특성이 매우 불규칙적이고 필기자나 쓰여진 언어에 따라 다양하게 나타나기 때문에 아직은 어려운 문제로 간주된다. 이를 해결하기 위하여, 본 논문에서는 문서영상을 텍스트라인 및 단어 단위로 나누는 새로운 방법을 제안한다. 제안하는 방법은 글자를 표현하는 방법으로 자획의 두께를 바탕으로 한 새로운 슈퍼 픽셀 표현방법과 이들의 특성을 바탕으로 한 에너지 최적화 방법으로 이루어진다.

본 논문의 요약은 다음과 같다. 첫 번째로, 새로운 슈퍼 픽셀 표현방법과 에너지 최적화 방법을 이용한 손글씨 문서의 텍스트라인 검출 방법을 제안한다. 다양한 언어로 작성된 문서에 대응하기 위하여 언어에 독립적인 텍스트 라인 검출 방법을 정규화한 연결 요소들(connected components, CCs)을 이용하여 개발하였다. 이 방법을 통하여, 제안하는 방법은 다양한 언어 및 필기 방식에 대하여 글자의 특성을 잘 예측할 수 있게 되었다. 이렇게 추정된 특성을 바탕으로, 최적화 결과가 텍스트 검출 결과로 나오게 되는 에너지 함수를 설계하였다. 실험 결과, 제안하는 텍스트라인 검출 방법은 다양한 손글씨 데이터베이스에 대하여 가장 좋은 성능을 나타내었다.

두 번째로, 고문서의 텍스트 라인 검출을 위한 전처리 방법을 제안한다.

현대에 작성된 손글씨 문서와 다르게, 고문서는 침출(bleed-through), 흐려진 글자 및 얼룩 등의 다양한 손상을 겪어서 문서의 품질이 좋지 않다. 이런 손상으로 인한 문제를 해결하기 위해, 이진화 방법과 노이즈 제거를 포함한 전처리 방법을 본 논문에서 제안하였다. 고문서의 이진화를 위해서는, 전역적 쓰레스홀딩(global thresholding) 방법과 지역적 쓰레스홀딩(local thresholding) 방법이 결합되어 얼룩이나 흐려진 글자에 대응하도록 하였다. 또한 텍스트라인 검출을 위한 에너지 최적화 방법도 고문서에 특성에 맞도록 수정하였다. 두 가지의 손상된 고문서 데이터베이스에 대한 실험 결과, 제안하는 전처리 방법을 이용한 텍스트라인 검출 성능이 기존 전처리 방법을 이용한 것에 비하여 좋은 성능을 나타내는 것을 확인하였다.

세 번째로, 스트럭쳐드 학습(structured learning) 방법을 이용한 손글씨 문서의 단어 분리 방법에 대해 제안한다. 본 논문에서는, 단어 분리 방법을 텍스트라인을 이루는 각 글자의 간격(gap)을 단어 내부 간격과 단어 간 간격으로 분리하는 문제로 보았다. 특히 손글씨 문서에서 나타나는 불규칙적인 특성들에 대응하기 위하여, 단어 분리 문제를 현재 간격의 특성뿐만 아니라 현재 텍스트라인에 간격들 간의 상호 관계를 고려한 이진 2차 문제(binary quadratic problem)로 나타내었다. 이 과정에서 많은 파라미터들이 필요하지만, 이 파라미터를 스트럭쳐드 서포트 벡터 머신(structured SVM)으로 학습하여 제안하는 방법이 사용자가 정의한 파라미터 없이도 작성된 언어나 필기 방식에 따라 잘 작동하도록 하였다. 두 가지 데이터베이스에 대한 실험 결과, 제안하는 방법이 다양한 언어의 단어 분리에 가장 좋은 성능을 나타내는 것을 확인하였다.

주요어 : 문서 영상 분리, 텍스트라인 검출, 단어 분리, 에너지 최적화 문제, 스트럭쳐드 학습, 슈퍼픽셀 표현
학 번 : 2009-20798